

ProWORDS/ProTERM
Word Processor/Smart Terminal
User's Manual
by
Larry Skutchan
Microtalk

ProWords File
ProWords Library
APW

ProWORDS/ProTERM

Word Processor/Smart Terminal

User's Manual

by Larry Skutchan

Copyright © 1985-1992
MicroTalk

Published by

American Printing House for the Blind

American Printing House for the Blind
1839 Frankfort Ave.
Louisville, KY 40206
(800) 223-1839

All Rights Reserved

04/96

Trademarks

The following titles are trademarks, registered trademarks, or copyrights of the indicated company.

Apple Computer, Inc.	Apple, Apple IIgs, Filer ProDOS
Applied Engineering	TimeMaster, RamWorks
RC Systems, Inc.	SlotBuster, DoubleTalk, ProSCAT
Sensible Software, Inc.	Sensible Speller
Street Electronics Corporation	Echo, Textalker
United Software Industries	ASCII Express

License

The enclosed software is licensed to customers for their use only with the terms listed below.

Terms

The American Printing House for the Blind (APH) agrees to grant to you a license to use the enclosed software with the terms and restrictions set forth in this agreement.

The enclosed software and its documentation are copyrighted by MicroTalk, and published by APH. You may not copy or reproduce this material except as specifically set forth in this license agreement.

1. You may make copies of this software for your own use provided you reproduce all copyright notices.
2. You may not distribute copies to others. You may, however, transfer the original and all copies, along with this license, if you provide APH with written notice and the transferee fills out a registration form and agrees to be bound by the terms set forth therein.

LIMITATIONS

This software is sold as is. Microtalk and APH assume no liability for damages of any kind that may arise from the use of this software and make no claims about its fitness for any particular purpose.

Apple Computer makes no warranties either express or implied regarding the enclosed computer software package, its merchantability, or its fitness for any particular purpose. The exclusion of implied warranties is not permitted by some states. The above exclusion may not apply to you. This warranty provides you with specific legal rights. There may be other rights that you have which vary from state to state.

Table of Contents

PART I: Introduction	1
Welcome to the Professional Family of Software	1
About this Manual	1
Conventions Used in This Manual	1
Manual Changes	2
What You Should Know	2
System Requirements	3
Overview	3
Getting Started	4
 PART II: ProWORDS	5
INTRODUCING ProWORDS	5
Tutorial	6
Indenting Paragraphs	6
Turning off Automatic Indenting	7
Proofing your Letter	7
Learning Cursor Movement Subdtiles	7
Correcting Mistakes	8
Printing the Letter	9
CURSOR MOVEMENT	10
By Character	10
By Words	11
By Sentences	11
By Paragraphs	11
By Screens	11
Vertical Movement Adjustment	12
To Beginning/End of Text	12
More Cursor Movement	12
To Beginning of Sentence	12
Finding Specific Text	12
Go Forward Indefinitely	13
Read Ahead	13
Skimming your Work	13
Cursor Movement Summary	14
EDITING	14
Overtyping	15
DELETING TEXT	15
Delete One Character	16
Delete Words	16
Delete a Block of Text	16
Delete to End of Text	17
Delete Forward to Specified Character	17
Delete to Previous Carriage Return	17
The Keystack	18
Recovering Deleted Characters	18
Purge the Keystack	18
MORE COMMANDS	18
Marks	18
Mark One	18
Setting Marks	19

Finding Marks	19
Move and Copy	20
Tabs	20
Clearing all Text in Memory	20
Search and Replace	21
Setting the Limits of the Search and Replace	22
Including Control Characters	23
The Wild Card	23
Key Popping	23
Voice the Keystrokes	23
The Sticky Space	24
Insert the Current Date or Time	24
Quit	25
Grade II Braille Translation	25
INFORMATION AND UTILITY COMMANDS	26
Pseudo Lines	26
Line-Oriented Material	27
Calculate Printing Position	27
Free Memory	28
Upper Case Detect	28
Toggle Case	29
ASCII Code	29
Preserving Defaults	29
Macros	29
Executing Macros	30
Defining Macros	30
Structure of the Macro File	30
The Parts of a Macro	30
The Macro definition	31
The Final Delimiter	31
Macro Example	31
Entering Apple Commands into your Macros	31
Table of Supplied Macros	32
Adding to the Supplied Macros	33
DISKS AND FILES	33
Table of Disk Commands	34
Saving Your Text	35
Block-Save	36
Loading Files	36
Overflow-load	37
Insert a File	38
Appending Files	38
Receiving Information from Other Devices	38
Table of Serial Commands	40
Transmitting Text to Other Devices	42
PRINTING	42
What are Directives?	42
The Format of a Directive	42
Changing the Format of your Document	43
Initiating and Controlling the Printout	44
Pausing a Printout	44
Cancelling a Printout	44

Notes About Buffers	45
Normal Directives	45
Printer slot: pr 1	45
Left Margin: lm 12	46
Absolute vs Relative	46
Right Margin: rm 75	46
Top Margin: tm 6	47
Bottom Margin: bm 6	47
Page Length: pl 66	47
Page Number: pg0	47
Spaces to Skip: sp0	47
Paragraph Indent: pp 0	47
Titles, Headers, and Footers: ti 1-7,0	48
Default Title Margins: dt0	48
Odd Titles: ot0	49
Page Numbering	49
Number Braille Pages: nb 1,0	49
Number pages with Roman Numerals: nr0	50
Left Justify Text: lj	50
Right Justified Text: rj	50
Center Justified Text: cj	50
Center Alignment: tc	50
Full Justified Text: fj	50
Skipping Lines: sk0	51
Command Character: cc	51
New Page: np	51
Odd Page: op	51
Conditional Page: cp 0	51
Pause: ps no message	51
Continue: cn no	51
Chain Files: ch <pathname>	52
Single Sheets: ss 0	53
Set Tabs: st 0	53
Contents Item: ci 0	53
Generating the Table of Contents	53
Emulation Mode: em 0	54
Force Upper Case: uc 0	54
Orphans and Widows: or 1	54
Inclusive Directives	54
Setting Printer Control Codes	55
Begin Underlining: bu	56
Finish Underlining: fu	56
Begin Boldface: bb	56
Finish Boldface: fb	56
Begin Subscript: b-	56
Finish Subscript: f-	56
Begin Superscript: b+	56
Finish Superscript: f+	56
Begin Italics: bi	56
Finish Italics: fi	56
Tab to Next Tabstop: tb	56
Tab to Right Margin with Specified Character: tr	57

Printing in Grade II Braille	57
Specifying Braille	60
Changing Standard Values	61
Braille Oriented Directives	62
Print in Braille: pb	62
Braille Numbering: nb 1,0	62
Suppress Multiple Carriage Returns: sc	63
Setting the Token Value: tk 2 or 5	63
Inclusive Braille Oriented Directives	63
Do Braille Only: db	64
Do Print Only: dp	64
End Do Section: ed	64
Do Single Braille Cell: ds	64
Grade or Level of Translation: gr	64
The Emphasis Symbol: bu	64
Transcriber's Option: to	65
English Braille Mode: eb	65
 PART III: ProWORDS Reference	 66
 PART IV: ProTERM	 70
INTRODUCING ProTERM	70
Setting Up	70
Notes About the Super Serial Card	70
Getting Started	71
Note to //c Users	71
TERMINAL MODE	71
Going Online	71
Commanding the Program	72
Commanding your Modem	72
Preparing to Establish the Connection	72
Establishing the Connection	73
Working Online	73
Setting Parameters	74
Communications Port	74
Baud Rate	75
Data Bits	75
Stop Bits	76
Parity	76
Half Duplex	76
Keystrokes Voiced	76
Incoming Data Clicking	76
Mode for Ham Radio	76
ASCII Upload Prompt	77
Outgoing Line Length	77
Incoming Line Length	77
Terminal Line Length	77
Returns Delayed	78
VT100 Emulation	78
Keypad Keys	78
PF Keys	78
Add Linefeeds	78

Set Clicker Volume on GS	78
Characters to Omit	78
Cancel Omitted Characters	79
Scrolling	79
Stopping and Starting the Remote System	79
Quit	79
SPECIAL SPEECH FEATURES	79
Silence Speech	79
A Note About Keyboard Buffering on the IIgs	80
Dumping the Speech Buffer	80
The Arrow Keys	80
Receiving Files	80
Receiving ASCII Files	81
Manually Saving the Buffer's Contents	82
Initializing the Buffer	82
Turning off the Capture Buffer	82
ASCII VS Binary	82
Summary of Program Transfer	82
XModem	83
CRC	83
The 1K Option	84
Binary II	84
ProDOS	85
TRANSMITTING FILES	85
Cancelling a File Transfer	85
Altering a Filetype	85
Macros	86
Structure of a Macro File	86
Structure of a Macro Definition	87
The Macro Access Character	87
The Delimiter	87
The Macro Action and Command Section	87
The Final Delimiter	87
The Macro Command Language	88
Using APPLE and OPTION Commands in Macros	88
Example of a Macro Using the APPLE Key	88
Calling Other Macros from within a Macro	89
Waiting for a Series of Prompts and Finishing the Waiting	89
Format of Waiting	89
Example of a Wait Sequence	90
Branch Points	90
Jump to Branch Point	91
Printing Messages on the Screen	91
Slight Pause	91
Combining the Parts	91
Another Waiting Sequence	92
CompuServe Logon Macro	93
The Supplied Macros	94
Monitoring a Macro	94
Issuing a Macro	94
Stopping a Macro	95

PART V: ProTERM Reference 96

PART VI: Appendixes 97

Appendix A: Background 97

Appendix B: Utilities 103

Appendix C: Pathnames and the Selector 108

Appendix D: Speech 116

Appendix E: MODEM COMMANDS 118

Appendix F: TROUBLESHOOTING 121

Appendix G: DAVEX, Talking Command Interpreter for
ProDOS 123

Handwritten notes in the bottom right corner, including the word "Hypothesis" and other illegible scribbles.

PART I: Introduction

ProWORDS/ProTERM

Preliminary Material

Welcome to the Professional Family of Software

Thank you and congratulations on choosing ProWORDS, ProBRAILLE, and ProTERM. You should find these packages the most powerful available. You'll find, too, that they're a pleasure to use. The professional family is designed to be simple to learn yet able to meet your needs for years to come.

About this Manual

This manual is divided into six distinct sections. The preliminary material, such as this introduction, comprises section one. It covers the basics of what you need to know to use this software, what hardware is required, and basic definitions of terms. The second part of the manual contains the complete ProWORDS operating instructions. The ProWORDS reference card is in Part III. ProTERM's detailed command guide to operations is contained in the fourth section of the manual. The ProTERM reference card is in part V. The sixth section of the manual, the appendixes, covers general concepts that are common to both ProWORDS and ProTERM. It also provides background information for new computer users. The ProBRAILLE User's Manual is supplied on the 3.5 inch manual disk and provides additional details and technical information concerning braille translation not covered in this manual.

Conventions Used in This Manual

In order to convey the information about keystrokes, command keys, and other command information in a reliable and meaningful way, certain conventions have been adopted. They are as follows.

Command Key Representation Normally, its easy to instruct you to press the "A" key or press RETURN. Those keys are clearly labeled (and spoken) when you press them; they are denoted in this manual with all capital letters. ESC, for instance, is the upper left key. Those keys marked with symbols (APPLE OPTION) are represented by putting their name in all caps. LEFT, RIGHT, UP, and DOWN indicate the cursor control arrow keys on the bottom right end of your keyboard. Less clear are accurate ways to express multi-key combinations--that is a command that you enter by holding down one of the modifier keys while you type a letter.

Besides the obvious SHIFT--which already has a standard means of representation-- your Apple computer has three other modifiers; APPLE, OPTION, and CTRL. When a command requires you to hold down one of these modifiers while pressing another key, that command is represented with the name of that modifier, a hyphen, and the key to press with the modifier. APPLE-A, for example, means hold down APPLE and press "A" just as you would hold down SHIFT to make a capital "a". Let the hyphen clue you about the modifier.

Part of your success using the computer involves issuing command keys correctly. Remember that the modifier keys act like the SHIFT key; you hold down the modifier, then, while the modifier is still depressed, press and then release the other key. Finally, when your finger is completely off the other key, release the modifier key.

In this manual, "CTRL" represents the CONTROL key located left of the "A" key. "APPLE" represents the OPEN APPLE key immediately left of the SPACE BAR on the //e and the //c. The APPLE key is the larger, second key left of the SPACE BAR on the IIgs and new //e. "OPTION" is the key located immediately right of the SPACE BAR on the //e and //c and is sometimes called CLOSED APPLE. OPTION is located left of APPLE on the IIgs and new //e. Remember, each of these modifiers has no effect when pressed by itself. You must hold the modifier down while you press another key.

Manual Changes

There may be information about this software that has changed since the manual's publication. Such information is contained in a file on your disk called UPDATES. Be sure to check this file.

What You Should Know

Before attempting to use this software, you should already be familiar with the material in the owner's manual that came with your computer. You may, however, get much of the information needed to get started in the appendixes of this manual. It is strongly recommended that you read these manuals, however. They build a powerful base on which you will expand your computer knowledge and effective use.

While they are not designed as a substitute for Apple's manuals, the appendixes of this manual should get any new user well on the way to understanding the operation of the Apple computer. "Appendix A" covers some basic concepts about the microcomputer that should prove useful to the beginner. "Appendix B" details the usage of the utility programs supplied on the master disk. "Appendix C" explains how to use the selector to easily find files on your disks. It also covers disk operating commands which are common to both ProWORDS and ProTERM. Speech commands and syntax are discussed in "Appendix D". "Appendix E" discusses modem

commands and "Appendix F" outlines the answers to commonly asked questions. The DAVEX command interpreter is explained in "Appendix G."

System Requirements

ProWORDS, ProBRAILLE, and ProTERM all require an Apple //e (enhanced), //c, //c+, or IIgs computer with at least 128K of RAM and at least one disk drive. They are all more convenient and faster, however, with more memory and disk drives.

This software works with the following synthesizers: SlotBuster, DoubleTalk, Echo II, Echo+, Echo IIB, Echo IIC, Echo Commander, and Echo Cricket. If you have an Apple //c, you should use the Echo Cricket or Echo IIC synthesizer, otherwise, any of the others work with any Apple II with slots.

ProTERM requires additional equipment if you have a //e. You will need an Apple Super Serial Card (SSC) or a modem that emulates the SSC. The //c's and //gs already have serial ports.

Overview

The ProWORDS word processing system, ProBRAILLE translator, and ProTERM telecommunications package are "natural" to use. Wherever possible, commands are exactly what you expect. In ProWORDS, for instance, the four cursor movement keys always move your cursor; the DELETE key deletes from your text. Commands are arranged so they are easy to remember. They are arranged in three distinct groups, the cursor movement, editing, and disk commands. In general, cursor movement commands take effect when you press one of the cursor keys. Editing commands take effect when you use the CTRL key with a letter key, and disk commands are executed by holding down the APPLE key while striking a letter. In addition to the three basic command types, ProWORDS supports the use of macros by holding down the OPTION key while striking a letter. (Macros are command keys that you set up yourself to perform a series of other commands. They are fully discussed later in this manual.)

The Command structure ProWORDS and ProTERM employ is built around the "palette" concept. The advantage of using the palette system becomes apparent when you realize that every command is available at any time. There are, in short, no modes. When you want to use a function, you press the command for it.

ProWORDS, ProBRAILLE, and ProTERM are the result of thousands of hours of testing, coding, and evaluation. The design of the programs incorporates features for maximum efficiency, power, and ease of use with a minimum of keystrokes and unnecessary talking. MicroTalk has worked hard to bring you an easy-to-learn,

easy-to-use, powerful word processing system and telecommunications system designed for use with speech synthesis.

Getting Started

Before you begin using this software, make a backup copy of the program disks. You can use any disk copy utility for this. The programs are not copy protected. They are, however, copyrighted and duplication for any other purpose is strictly prohibited. Once you make the backup copies, store your original disks in a safe place and use the copies.

To begin using this software, place the master disk into your boot drive and turn on the computer. The disk spins for a few seconds, and the following screen appears.

```
Words Spell Term Filer Grade2 Basic or Help
```

```
/user/util:
```

This title screen shows two things; a menu line and the name of the currently opened folder.

The menu items are names of programs to use. You start any of them, or any other program, by typing its name.

You don't necessarily have all the programs listed on the menu. The "Spell" option is designed to use the Sensible Speller available from APH. You can customize the menu to your taste by editing the file MENU.TXT,.

If you try to use a program that isn't on your disk, you'll see a message like this:

```
huh?
```

The second item of information shown on the main screen is the name of the currently opened folder. (A complete discussion on folders is found in "Appendix G.")

If you are interested primarily in ProTERM, you should skip to the "ProTERM" section of this manual now.

PART II: ProWORDS

Professional Talking Word Processor INTRODUCING ProWORDS

Selecting WORDS from the menu starts the word processor. As soon as it beeps, you have an electronic workspace in which to work and you may begin using it immediately. To begin writing, just start typing; it is that simple. If you make a mistake, use the DELETE key, at the upper right of your keyboard, to delete it.

As you type, notice that ProWORDS automatically moves words from one line to the next when you reach the end of the screen. Do not press the RETURN key when your cursor reaches the screen's edge, let the computer do this work for you. Use the RETURN key, instead, to mark paragraphs or places where you definitely want to start a new line.

When you've written a few sentences, file your work. To file your work (or to store it permanently), press APPLE-S.

When you press APPLE-S, ProWORDS says "SAVE" and then, "TYPE FILE NAME." The name should suggest the contents of your text. That makes it easier to find the document in the future. If you wrote a letter to your mother, you might call it MOM, or if you wrote a report on the annual rainfall in the jungles of Peru, you might call it RAIN or PERU.

When you type the name and press RETURN, your disk spins as ProWORDS stores the information from the workspace onto your disk.

Once you store the text on a disk, it is filed. To make sure your work got on the disk, press APPLE-QUESTION MARK. This shows a list of the files in the open folder. The file you just stored should be at the bottom of the list. To move your pointer quickly to it, press the UP ARROW once. Notice the pointer jumps from the top file on the list to the bottom one. If you wanted to load that file back into your workspace, you would press the RETURN key. There is, however, no need to do so now because the original text is still in your workspace. Press the ESCAPE key, at the upper left of your keyboard, to return to it. When you stored the information, ProWORDS made a copy of the text in your workspace and put it on the disk; the original material is not altered in any way.

Each disk holds only so much information (about 140,000 characters on a 5.25 and 800,000 on a 3.5 disk). You must, therefore, eventually use other disks. Before you can use a new disk, however, you'll need to "format" it. Formatting personalizes the disk to the Apple computer. (Disks may be used on nearly any brand.) To format a disk, quit ProWORDS and use the "init" command

or FILER. (You can get complete information on the init command by typing "?init" from Davex's folder name prompt or use Filer's "?" command or Appendix B of this manual.)

Tutorial

Assume you want to write a letter to APH to inform us of your thoughts about the ProWORDS system. This section of the manual discusses the step-by-step procedure for such a task.

First, of course, start the word processor.

Once the word processor is running, you may begin typing immediately. You probably want to begin the letter with your return address. Type it now. Remember, if you make a typing mistake, use the DELETE key to correct it. Press RETURN at the end of each line of the return address because you definitely want each element of the address to print on a separate line. Notice how ProWORDS makes your speaker sound with a distinctive "bonk" with each press of the RETURN key. You can use this sound to identify carriage returns in your work. On yet another line, you may wish to type the date.

If your computer has a clock/calendar or if you manually set the date when starting the disk, press APPLE-GRAVE ACCENT to automatically type the date.

After typing your return address and the date, press RETURN two or three times to leave some blank space before typing the inside address. When you finish the inside address, press RETURN three more times. This ends the last line of the inside address then leaves two blank lines for spacing.

Type the greeting and three more carriage returns. This leaves two more blank lines before the body of the letter.

Indenting Paragraphs. ProWORDS provides a means of automatically indenting paragraphs. To do this, you'll use a "directive" to tell the computer how many spaces to indent each paragraph.

If you want to indent paragraphs within your letter, type ".pp#" and press RETURN. You just typed a directive. Directives are commands that tell the program how to format your text when it is printed. Directives do not get printed, nor do they cause an extra line to print on your work. They simply provide a flexible means of controlling the format of your printout. The pp directive is used to control the amount of indention that occurs at each paragraph. (Paragraphs are indicated with the carriage return.) You only type the pp# once. The program will indent automatically when you press RETURN. The number sign you included in the directive tells the program how much to indent the paragraph. It means that if you are printing in braille, each paragraph is

indented two spaces; if printing in ink, each is indented five spaces. If you want to use a different number, substitute the number sign for the actual number to indent or use the tk directive to change the values associated with the NUMBER SIGN in the pp directive. You can read about each of the directives in the "Directives" section of this manual.

Now, simply type your letter pressing one or two carriage returns between paragraphs depending on whether or not you want a blank line to separate each paragraph. If you want a blank line, press RETURN twice--once to end the line and once to leave a blank line. If you don't want blank lines between paragraphs, press RETURN only once to end the line and begin on a new one.

When you complete the body of the letter, type three carriage returns. This, again, ends the current line and leaves two blank lines in the printout.

Turning off Automatic Indenting Once the body of the letter is complete and you have skipped the appropriate number of lines to separate the closing from the body of the letter, you will want to turn off the indenting directive. (You want the closing to line back up with the left margin.) To turn off indenting, type ".pp0" followed by a carriage return. After you do that, new paragraphs (and lines) remain lined up with the left margin.

After turning off paragraph indentation, type your closing, five carriage returns, and your name. The five carriage returns leave room for your signature. You could substitute ".sk4" for the returns to accomplish the same thing. ".sk4" is another directive. Instead of telling ProWORDS how to indent paragraphs, the sk directive tells ProWORDS to skip lines on the print out. The number following the "sk" in the directive tells it how many lines to skip. In this case, pressing four carriage returns and typing .sk4 do the same thing.

Proofing your Letter When you make your initial draft of the letter, look it over for style and syntax. Use the ARROW keys to move the cursor. As you press the ARROW keys, the cursor moves in the direction of the ARROW.

Learning Cursor Movement Subtiles Start by moving all the way to the beginning of the letter. Press OPTION-UP to do that. Your cursor should be at the beginning of all the text and in the top left corner of the screen. Now press the DOWN ARROW key. As you do, ProWORDS moves the cursor down by sentences or lines, whichever comes first. As you move down, notice that the cursor always ends up at the point where the speech stopped. If, for instance, you press DOWN ARROW on the first line, you heard your street address. If your address were "337 S. Peterson", you heard that and your cursor was left positioned immediately right of the "n" in "Peterson". There is a carriage return to your cursor's right.

You can confirm this by pressing RIGHT ARROW. The computer "bonks" telling you that your cursor just crossed a carriage return. That means it went to a new line. Another press of RIGHT ARROW would move the cursor over the first letter on that line. Instead of pressing RIGHT ARROW, though, press LEFT ARROW and move the cursor back to where we left it. Notice the "bonk" again. Your cursor just crossed that carriage return again and is now again positioned to its left and right of the "n" in "Peterson".

When you press DOWN ARROW, ProWORDS moves the cursor (and announces the text) with these rules: The first character is always skipped. That lets you press DOWN ARROW while your cursor is at the end of a line or paragraph; the cursor skips that initial carriage return. The cursor continues moving forward through the text until it reaches either a carriage return (beyond the initially skipped one) or two consecutive spaces. Your cursor is always left positioned immediately right of what you just heard and there will always be either a carriage return or two or more consecutive spaces right of the cursor (that's how the cursor knows where to stop.)

The rules cursor movement follows when you press UP ARROW are a little more complicated. When you press UP ARROW, ProWORDS scans to the cursor's left until it finds either a carriage return or two consecutive spaces; it then goes back another sentence from there, reads that sentence, and leaves your cursor positioned at the end of it. This gives the effect of making the UP ARROW read the previous sentence and leave your cursor at the end of it. The sentence you were working with is now two spaces right of the cursor and you can position yourself in that sentence with APPLE-RIGHT.

By combining the convenience of sentence by sentence movement and the fact that the speech is silenced when you press the ARROW keys, you'll be able to quickly proofread your work without being constricted to the confines of lines on a computer screen. Instead, your movement follows the flow of the English language and you'll quickly see a much better picture of the overall work.

UP ARROW and DOWN ARROW can move your cursor a pretty good distance through your text. Sometimes, you'll want finer cursor control than that. Use APPLE-RIGHT to move through your text a word at a time. APPLE-LEFT moves back a word at a time.

Correcting Mistakes If you find a sentence or passage that needs correcting, delete the bad text and re-type the text like you want. Zero in on the text that needs editing with APPLE-RIGHT and APPLE-LEFT.

There are a number of ways to delete text. If the mistake involves only a letter or two, position the cursor to that letter's right, then press DELETE. Once you delete the letter or letters, type the new ones.

When you move the cursor by letters, ProWORDS announces the character the cursor crosses. The cursor, in ProWORDS, is always positioned between two characters. As you move it right, the cursor crosses the letter to its right and pronounces it. If you then press the LEFT ARROW, the cursor moves back to the left of that character, again pronouncing it as the cursor crosses it.

Printing the Letter Before you print the document, you might like to know if it all fits on one page or if, perhaps, your name slipped over to the second page. Press APPLE-K. Pressing APPLE-K calculates the format of your work according to the directives you have used. The command shows you what the printout looks like on the screen and tells you the position of your cursor on the printout. This position is expressed as three numbers separated with periods. If the cursor were on the first page, the 54th line, and the 27th column, the indicator would be "1.54.27."

If you find that you just barely crossed the page boundary, you have several choices on how to proceed. You may, for example, extend the right margin to allow a few extra characters per line. This may be enough to allow the letter to fit onto one page. To extend the margin, move the cursor back to the beginning of your letter (using the OPTION-UP ARROW command) and type ".rm+3" and a carriage return. Move back to the end of the file with the OPTION-DOWN ARROW command and try the APPLE-K command again. If you still don't have room, consider making the left margin a few spaces smaller. Move back to the top of the file and type ".lm-3" and a carriage return. Now move back to the bottom and check it again. If there is still not room, you may consider deleting the two margin directives you just added and extending the amount of space between your return address and the inside address. This will force more text to appear on the second page. You may also want to increase the size of the top and bottom margins which will also force more text to appear on the second page.

Once you decide about the margins, save your work. Press APPLE-S and type a name for your letter. Use APH as the name for the text. Press RETURN after typing your letter's name. The disk should spin as a copy of your letter is permanently filed on disk. Once a copy of your text is safely filed away on your disk, print it using the APPLE-P command. Be sure, however, that your printer is prepared. It should be on and the paper should be set so that the printer's head is just below the perforation between the pages.

Note: if your printer is connected to a slot other than slot one, you need to add yet another directive to the beginning of your work. Move to the top of the file and type ".pr", the slot number of your printer, and a carriage return. This directive tells ProWORDS to send the printout to a different slot. (There are some "special" slots assigned for use with the pr directive which you can explore in the "printing section of this manual.)

You will always want, of course, to proof read and edit your work before sending it. You may also want to use the SENSIBLE SPELLER to make sure there are no typing mistakes or spelling errors before printing. The following sections show you how to use the cursor movement commands to position the cursor for editing.

CURSOR MOVEMENT

There are two reasons for moving the cursor. First, moving the cursor lets you see what text is in the computer's memory. As you move the cursor, the text is displayed on the screen and spoken by the synthesizer. The second reason for moving the cursor is to position it for editing your work.

ProWORDS provides a range of cursor movement commands. Since you use the cursor to point at the spot where you wish to edit, you want to control the cursor in two ways--direction and distance.

The direction of cursor movement is controlled by the arrow key that points in the direction you want to move. (The arrow keys are located at the bottom right of the keyboard.) The keys are (from left to right) LEFT ARROW, RIGHT ARROW, DOWN ARROW, and UP ARROW.

Control the cursor's distance with the apple keys. Use the appropriate apple key as if it were a SHIFT key while you press one of the cursor movement keys. The APPLE acts as a medium-range cursor distance shift. The OPTION key acts as a long-range shift.

By Character

Use the LEFT and RIGHT ARROW keys to move your cursor around for character-by-character editing. The RIGHT ARROW moves the cursor one character right. The LEFT ARROW moves it one character left. (Note that you do not use the LEFT ARROW to correct mistakes; use the DELETE key to delete the character immediately left of the cursor.)

Notice how the cursor always splits your text to make room for itself. If you type a letter, ProWORDS inserts the character into your text at the cursor. The cursor then moves to the next position. If you move right or left with the cursor keys, the cursor always crosses a letter.

Each character the cursor crosses gets pronounced by the synthesizer. If that character is a capital letter, the pitch of the pronunciation is raised to alert you of this fact. If the character is a control character, it is pronounced as "CONTROL" then the letter of that control character. (Control characters are displayed on the screen as inverse letters.) CTRL-C, for instance, is pronounced "CONTROL C" and is displayed as an inverse "C".

One control character, the carriage return, gets special treatment. When your cursor crosses a carriage return, ProWORDS alerts you with a tone. The carriage return is actually a CTRL-M, but it would be annoying to hear "CONTROL M" and even more distracting to see an inverse "M" for every one of these rather frequently-used characters. A complete discussion of the use of the carriage return appears in the "Editing" section of this manual.

ProWORDS alerts you when you try to move the cursor past the beginning or end of the text in memory. If, for example, you pressed the LEFT ARROW when your cursor were already at the beginning of the text, ProWORDS buzzes.

By Words

APPLE-LEFT and APPLE-RIGHT move by words. If there is more than one space between the words, your computer's speaker is clicked for each space past the first one. This lets you know when there are extra spaces in the text. As you move to a new sentence, you'll hear one click indicating one extra space which is, of course, normal for sentences.

By Sentences

UP ARROW and DOWN ARROW move by sentences. Pressing DOWN ARROW moves the cursor and pronounces the text from the cursor to the end of the sentence. The cursor ends up, as always, where the speech stops; immediately after the period ending the sentence.

UP ARROW moves the cursor to the end of the previous sentence. The previous sentence is spoken.

Note: ProWORDS' ability to use sentence structure depends on the fact that your sentences end with two spaces. Make sure you space twice after each sentence.

By Paragraphs

If you hold down the OPTION key with the LEFT or RIGHT ARROW, your cursor jumps backward or forward respectively to the next carriage return in the text. This means, in most cases, that the OPTION-LEFT or OPTION-RIGHT commands move the cursor to the beginning or the end of a paragraph. (See more about paragraphs in "Editing".)

By Screens

As with the LEFT and RIGHT ARROW keys, you can use the UP and DOWN ARROWS with the APPLE key to change the amount of text to skip. APPLE-UP ARROW moves the cursor up your page about half a screen; APPLE-DOWN ARROW moves the cursor down about half a screen. The actual amount of vertical cursor movement is adjustable.

The APPLE-UP and APPLE-DOWN commands allow you to quickly scan through large bodies of text because, as you move with APPLE-UP or APPLE-DOWN, the text is not pronounced by the synthesizer.

Vertical Movement Adjustment

To adjust the number of lines your cursor moves up or down the screen with the APPLE-UP and APPLE-DOWN commands, hold down BOTH the APPLE and OPTION keys and press the UP ARROW to increase the movement or the DOWN ARROW to decrease it. The current "increment" value, expressed as the number of lines skipped, appears in the upper left corner of your screen. The range of the increment value is from 1 to 255.

To Beginning/End of Text

The UP and DOWN ARROWS can be used to put the cursor directly at the beginning or end of the text in memory. Use the OPTION as if it were a SHIFT with the UP ARROW to put the cursor at the beginning of text. Use OPTION-DOWN ARROW to move the cursor directly to the end.

More Cursor Movement

In addition to the cursor keys, you may use several control commands to move the cursor. The following cursor movement functions all use control commands.

To Beginning of Sentence Press CTRL-B to make the cursor jump instantly back to the first character on the current sentence.

Finding Specific Text You may move your cursor so that it moves to the next occurrence of any text you wish. Press CTRL-F then type the text to find.

After Pressing the CTRL-F and typing the text to find, you have a choice about where to begin looking and which way (from the cursor) to look. If you press RETURN after typing the text to find, the search begins at the beginning of your document regardless of the cursor's position. To find the text that matches your specifications only beyond the cursor, press DOWN ARROW after typing the text to find. To search from your cursor back, press UP ARROW after you type the text to find.

When you use the find command, ProWORDS moves your cursor directly to the next occurrence of the text in the direction you specify. To find another occurrence of the same text, press CTRL-F again. This time, ProWORDS types out the last thing you looked for. To find the next occurrence of that same text, press the direction indicator key as previously described. You don't have to type the text again.

ProWORDS moves your cursor directly to the specified text even if it does not match the case of the text you typed. The find command, in short, is not case sensitive. This means that if you want to find the word "hello", typing CTRL-F followed by "hello" finds "Hello", "hello", and every other combination of capital and lower case letters. If you have to find text that exactly matches your specifications including the case of the letters, use the search and replace command discussed in the "Search and Replace" section of this manual.

If the text you specify does not appear in your document, ProWORDS tells you that it could not find the text and leaves your cursor in the position where you first issued the command.

It does not matter if the text you are looking for is contained within another word or if it is a part of another word. You may search for more than just a few letters, words, or phrases. You may also search for control characters. Simply precede each control character to find with a press of the CTRL-O key. Finding the word "Hello" followed by a carriage return is done by pressing CTRL-F, typing hello, pressing CTRL-O and RETURN, then pressing the direction indicator key. Finding the next instance of that text is done by pressing CTRL-F followed by the direction indicator key.

Go Forward Indefinitely If you press a CTRL-G, your cursor jumps ahead by words until you press another key. This feature is handy for those using speech synthesis because when a key is pressed, the cursor stops instantly. If you do not press a key the "go ahead" command continues going forward until the end of your text.

Read Ahead In addition to moving ahead by words, you may use CTRL-R to read and move your cursor by sentences. To quit reading by sentences, press any key.

The read-by-sentences operation provides an excellent means of reviewing your work for its overall content because the sentences are inflected correctly and the synthesizer pauses properly at the end of each sentence. You may wish, during this operation, to change the synthesizer's punctuation mode to "some" for an even more "literary" reading is achieved. Look in "Appendix D" for details on changing your synthesizer's punctuation level.

If you are using the read-ahead (CTRL-R) function for proofreading, you will notice that picking out extra spaces is fast and easy. The synthesizer pauses when two or more spaces are encountered. It is this pausing that provides the "natural" sound of the sentences when reading in this mode.

Skimming your Work As you use the read ahead command, notice that pressing DOWN ARROW makes the word processor jump immediately to the next sentence. This permits an efficient means of skimming your work; once you hear the noun and verb in a sentence, press

DOWN ARROW to skip immediately to the next sentence and repeat until you spot a group of sentences that all start with "It was" or "It has".

Cursor Movement Summary

You have probably noticed that the APPLE key and the cursor movement keys--used to control the cursor's distance--have been organized in a special way. The LEFT ARROW by itself, for instance, moves the cursor left one character; APPLE-LEFT ARROW makes it jump by words; and OPTION-LEFT ARROW pulls the cursor back to the beginning of the paragraph.

UP ARROW scoots up one sentence; APPLE-UP ARROW moves up by screens; and OPTION-UP ARROW slides the cursor back to the top of the text.

The APPLE enhances the arrow keys by pushing their capabilities a little further; the OPTION makes the cursor go as far as it can.

EDITING

ProWORDS inserts everything you type wherever your cursor is. This is a very powerful feature. The ability to always insert means that you simply type what you want without worrying about overtyping some text previously in memory. It is much easier to delete the parts you do not need anymore. (This is what computers are all about.)

When you insert a word, paragraph, or even an entire file, there is no need to reformat the text. Your text is stored internally as a continuous stream of characters and does not get formatted until print time. You may, however, have noticed that the display looks as if it has been formatting your text all along. In a way, it has. ProWORDS does not, however, insert carriage returns into the text as many older word processors have done. Instead, ProWORDS keeps track of the lines internally so that you are presented with the illusion of working with a line or screen editor yet you may utilize the power and flexibility of paragraph orientation.

The Carriage Return

When a word processing program is paragraph oriented, it means that the software uses carriage returns to separate paragraphs. You never need to press RETURN, therefore, until you are ready for a new paragraph. When you press the RETURN key, ProWORDS inserts a carriage return into your text. Many people like two carriage returns between paragraphs. That leaves a blank line between them.

Never press the RETURN key when your text reaches the screen's right edge unless you definitely intend to end a line there. The program will automatically adjust words so that they don't break on

the screen. Pressing a RETURN at the end of a screen line forces the word processor to begin a new paragraph. Use the RETURN only to indicate either a new line or a new paragraph.

When it is time to print the file, ProWORDS formats each line so that it begins at the twelfth column on the paper and it makes sure that no character prints past the seventy-fifth column. That is, of course, unless you indicate otherwise. You may change these margins and a variety of other formatting details with the use of printing directives which are covered in the "Printing" section of this manual.

Overtyping

Pressing APPLE-TAB selects "overtyping mode". We do not recommend that new users use the overtyping mode. It is provided as a feature the advanced user should appreciate.

Overtyping mode overtypes the character under the cursor instead of squeezing that character out to the right. This feature is used in applications that require an exact, preset format. You might, for instance, use the overtyping mode when filling out forms. If two items appeared on the same line of the form, you use overtyping to fill in the first item without making the second item scoot down the line.

There are two ways to return to insert mode. If you continue typing until the cursor reaches the next carriage return, ProWORDS temporarily re-enters insertion mode so you do not type past a paragraph boundary. You may also switch back to insert mode by pressing APPLE-TAB again.

You will probably not need the APPLE-TAB command. Normally, rather than overtyping text, you will want to remain in insert mode and use one of the many powerful methods of deleting text to get rid of what you do not want. Instead of toggling into overtyping mode, for instance, to type over the last few characters of a sentence, try moving to the first character where you would normally overtype then insert the replacement text. When you have made the addition, use the CTRL-D, delete to specified character, command followed by a period to delete the text from the cursor right to the next period. You may read more about the CTRL-D and a wide variety of other deleting techniques in the following paragraphs.

DELETING TEXT

ProWORDS supports a wide array of deleting commands. They are as follows:

Delete One Character

The DELETE key, at the upper right of your keyboard, removes the character immediately left of the cursor and saves it for you in a temporary holding place called the "keystack". To recover characters from the keystack and insert them into your text at the cursor, use the APPLE-SPACE BAR command. You can actually delete a sentence, move your cursor, and then put that sentence wherever you want.

Note that the LEFT ARROW does not delete a character from your text; it moves the cursor left one character. Use the DELETE key to delete the character immediately left of your cursor. (Remember, you can always get it back if you want it.)

Delete Words

Use the DELETE key with the APPLE to delete the word left of the cursor from your text. The APPLE-DELETE command saves the deleted word on the keystack so you can use the APPLE-SPACE BAR command to recover it one character at a time or the OPTION-SPACE command to recover text from the keystack a word at a time.

The delete-word command works by deleting characters to your cursor's left, one at a time, until the cursor comes to either a space that wasn't preceded by another space, a carriage return, or the beginning of the text. That means that if there are multiple spaces between the word and the cursor, both the spaces and the word are deleted.

You may not use the delete-word command to delete past a carriage return paragraph boundary. Use the delete key by itself for this. This "safty valve" keeps you from accidentally deleting more than you intend.

Delete a Block of Text

OPTION-DELETE deletes the block of text between your cursor and mark one. Unless you've changed it, mark one will be at the beginning of your text. By changing mark one with OPTION-1, you specify the beginning point of the text to delete. Mark one represents the beginning of the block to delete; your cursor indicates the bottom of the block.

When you press OPTION-DELETE, ProWORDS makes sure you really want to do this by asking, "Delete marked block?". If you're sure you want the block deleted, answer with "Y". Press "N" if you decide you don't want the text deleted or if you are in doubt about where mark one is set.

Delete to End of Text

APPLE-OPTION-DELETE deletes text from your cursor to the end of your work. When you press APPLE-OPTION-DELETE, ProWORDS asks "Delete to end?". If that's what you want, press "Y"; if you aren't sure, press "N" or ESCAPE.

Delete Forward to Specified Character

The DELETE key lets you delete the character to the left of the cursor. The CTRL-D permits forward deleting. This is one of the most used of the delete commands. You use it by pressing the CTRL-D then pressing another key. The key you press will be used as an indication of how far you want to delete forward in the current paragraph. If, for example, you press CTRL-D and then the SPACE BAR, ProWORDS deletes all characters from your cursor right up to and including the next space. This is, if it finds a space in the current paragraph.

The delete (CTRL-D) command will not cross the paragraph boundary (the carriage return) to find your specified character since this could be disastrous. You may, however, delete to the next carriage return by pressing CTRL-D and then pressing the RETURN key. This is an especially handy command to delete entire paragraphs.

The delete forward (CTRL-D) command is useful for deleting or moving sentences. Just move to the end of the previous sentence, press the CTRL-D, and then press a period. This command deletes the entire sentence. Remember, no reformatting is necessary. Since this operation also saves your deleted text in the keystack, you could move your cursor somewhere else then pull the sentence out of the keystack and insert it into your text at that point by using the APPLE-SPACE BAR command.

If you accidentally press a CTRL-D, you may cancel the command by pressing another CTRL-D.

Delete to Previous Carriage Return

If you hold down the APPLE key and press "X", all the text from your cursor left to the previous carriage return, including that carriage return, is deleted from memory and saved in the keystack. Remember, if you accidentally press the APPLE-X, you may use the APPLE-SPACE BAR command to recover characters from the keystack.

The APPLE-X command effectively deletes from your cursor left to the beginning of the current paragraph. If, on the other hand, you were editing line-oriented material, such as computer programs, the APPLE-X deletes the line back to and including the previous carriage return.

The Keystack

The keystack is a special place in your computer's memory that retains 256 of the last characters you deleted. On the 256th character, the keystack "wraps" around so that you always have available the last 256 characters you deleted.

Recovering Deleted Characters You may hold down the APPLE key and then press the SPACE BAR to recover the last character you deleted. Repeat this procedure to recover more characters. To recover entire words from the keystack, hold down the OPTION then press the SPACE BAR. The OPTION-SPACE BAR command pulls one word from the keystack by pulling individual characters out until it finds a space. If no space is found within the first fifteen characters, ProWORDS stops; so, the OPTION-SPACE BAR command recovers either one word at a time or fifteen characters at a time from the keystack.

Purge the Keystack Press APPLE-OPTION-SPACE to remove all characters from the keystack. You can use this command to clean the keystack before using it for a small move operation. For moves that require more than 255 characters, see "Copy and Move" later in this manual.

MORE COMMANDS

The following commands are not used directly to edit text. They help, however, by providing several functions that both increase flexibility and display information.

Marks

A most handy feature of ProWORDS is the ability to set nine different marks in your text. You might wonder what you would do with a mark if you had it. Actually, there are several good reasons for using marks.

The ability to set a mark at any point in your text allows you to immediately refer back to that point without having to search through the rest of your document. You may wish, for instance, to mark your current position and scan back to some previous information. You could, if you wish, then set another mark at this information for future reference. Now, switching between the two points requires only a single keystroke.

When the program starts, all marks are set to the top of your text. You may, however, change this condition. See the section entitled "Preserving Defaults" for more details on this procedure.

Mark One One of the marks, mark one, is special. It is used by several other commands in ProWORDS including saving a segment of

text, moving and copying operations, and deleting a segment of text.

Mark one is always used for deleting blocks of text. The procedure used to delete a block, as you have seen, is this:

1. Move your cursor to the top of the block of text to delete and set mark one with OPTION-1.
2. Move your cursor to the bottom of the block to delete.
3. Use OPTION-DELETE to delete the text between your cursor and mark one.

See the sections "Move and Copy", "Deleting a Block", and "Saving a Segment" for more uses of mark one.

Setting Marks To set a mark, just hold down the OPTION key and press the number of the mark you wish to set. Whichever number you choose, that mark will be assigned the current position of your cursor.

Marks work by making ProWORDS remember the exact point in memory where your cursor is when you set the mark. If you type text above your marks, the mark remains at the point where you set it but the actual text may seem to have moved down. To avoid potential problems, be sure to set the marks for move and copy, save segment, and block deletions immediately before performing the desired operation.

Since ProWORDS stores the marked positions directly in its internal memory, you may use this feature to accomplish some unique maneuvers.

Let us assume that you are in the middle of a long file and you need to refer to another file. You can set a mark and load the other file; then, when you find what you need, you simply reload the original file and jump directly to that mark. (Remember, if you are editing the first file, save it before loading another.) You could even set another mark for the second file so that switching between the two and finding the pertinent information happens quickly and painlessly.

Finding Marks Whenever you wish to move your cursor backward or forward to a desired mark, hold down the APPLE key and press the number of that mark. Your cursor instantly jumps to the desired mark without your having to decide what the relationship of the cursor is to the desired mark. If your mark is a long way from the cursor, it might take a second or two to get there. ProWORDS informs you with the familiar tone when the cursor reaches the mark.

Move and Copy

ProWORDS includes a powerful move and copy feature. It, like the rest of the commands, is simple and logical to use. It works like this:

1. Move the cursor to the top of the block of text to move or copy and set mark one.
2. Move the cursor to the bottom of the block and set mark two.
3. Move the cursor to the desired destination of this text and press CTRL-C (for copy). The marked text will be saved on the disk to a temporary file called "T" and then inserted at the point of your cursor in the desired location.
4. Next, ProWORDS will ask you if you wish to delete the original block of text. If you do, just press "Y" (for yes).
5. If you do not want the original block deleted from the text, press "N" (for no).

WARNING: If you have a file on your disk named "T", this command will destroy the contents of that file and replace it with the marked block of text. If you want to keep the file named T, use the APPLE-N, name-change command to change its name.

Tabs

Tabs are set in your text with the use of directives. Directives are characters that you type directly into your text but that do not get printed; instead, they change the way your work is printed. Using the tab directive, for example, sets the tab stops. When you use the TAB key, ProWORDS inserts yet another directive into your text that causes the printer's head to move to the next tabstop before printing resumes. Get complete details on using directives in the "Directives" section of this manual.

Clearing all Text in Memory

When you are ready to clear memory and start a new document, use the new command (CTRL-N). When you press CTRL-N, ProWORDS asks "NEW? " to make sure that this is really what you want to do. If clearing memory is really what you want, press "Y". If it is not, press "N".

Search and Replace

ProWORDS provides you with a powerful search and replace command. It is invoked with CTRL-S. The search and replace command lets you either search only or search and replace. You inform the program about which you wish to do by using delimiters. A delimiter is a character you type to separate the different parts of the command. It can be any character which does not appear in the specific characters for which you are searching or replacing. A good delimiter to use, unless you are looking for one, is the semi colon.

An understanding of the procedure for searching precedes discussion of searching and replacing, so here is the procedure for using the search command.

1. Press CTRL-S to initiate the command.
2. Type a delimiter. Remember, a delimiter can be any character not in the characters for which you are looking. If you were looking for the word hello, it could be any letter except "H," "E," "L," or "O."
3. Type the text to find. (Capitalization counts.)
4. Type the delimiter character again.
5. To search from the beginning of your document regardless of the cursor's position, end the search command by pressing the RETURN key. To search only from your cursor forward, terminate the command by pressing the DOWN ARROW. You can search from the cursor back toward the beginning of your document by ending the search command with a press of the UP ARROW.

A typical search might look like this:

```
;hello;
```

This command finds all occurrences of "hello". The delimiter used in this search is the semi colon but it could have been any character not in the word "hello".

Each time ProWORDS finds a match to your specifications, it pauses, shows you the occurrence, and asks if it should continue searching. If you wish to find another occurrence of the text, press "Y". If you press "N", the command is cancelled and your cursor rests at that occurrence. If you continue searching beyond anymore occurrences of the desired text, ProWORDS returns the cursor to the position where the search command was first invoked.

If you include yet another delimiter, the search command becomes a search and replace command. It works by replacing the text found between the first and second delimiter with the text between the second and third delimiter. (Note that such a construct permits the replacement of text with nothing.) Here is how it works:

1. Type the first delimiter.
2. Type the text to find.
3. Type the delimiter again.
4. Type the text to replace the text included between the first two delimiters.
5. Type the delimiter again and press the appropriate key (RETURN, UP ARROW, or DOWN ARROW) to indicate the direction and scope of the command.

A typical search and replace command might look like this:

```
;hello;goodbye;
```

This would replace "hello" with "goodbye".

If you indicate, by using the above construct, that you wish to search and replace, ProWORDS will ask you if you would like to automatically replace all occurrences of the text or if you would like to verify each one independently. To automatically replace all occurrences of the text, type "Y" to the "AUTOMATICALLY REPLACE?" prompt. If you would like to be aware of what is going on and want to verify each replacement before it happens, answer "N" to the prompt; each occurrence will be displayed and you will be requested to confirm the replacement. Answer "Y" to replace the displayed text, "N" to move to the next occurrence, or press the ESCAPE key to cancel the search and replace command and return your cursor to the place it was when the search command was first invoked.

Setting the Limits of the Search and Replace Sometimes it is desirable to limit the range of the search and replace. This lets you use the automatic replacement option of the search and replace command without worrying if there will be unforeseen consequences to text in other parts of your document. You can set the limits of your search and replace by setting mark eight to the top limit, moving your cursor to the bottom of the text of interest, and using the normal search and replace command (specifying, of course, to search from the cursor back by terminating the search command with UP ARROW. Look at the following short passage for a demonstration of this feature.

Now is the time for all good men to come to the aid of their party.

Say you want to replace all spaces with "X", but you only want to replace those from the word "time" on. Of course, in this brief passage, it would be simple enough to make the replacements by hand. for the sake of demonstration, though, move your cursor to the end of the word "time" and press OPTION-8 to set mark 8. Now move to the end of the passage and press CTRL-S to search and replace. Type the following search and replace sequence and terminate the command with UP ARROW.

```
; ;X;
```

The resulting text looks like this:

Now is the timeXforXallXgoodXmenXtoXcomeXtoXtheXaidXofXtheirXparty.

Including Control Characters Your search string may contain any control character, including carriage returns. Simply precede each control character with a CTRL-O. The CTRL-O, as discussed in the "Printing" section of this manual, tells ProWORDS that you want to include the next character you type in your text and not use it as a command. In other words, if you want to replace all carriage returns with "@", the exact characters to type would be semi colon, CTRL-O, RETURN, semi colon, the at sign, and finally the ending semi colon delimiter. Note that this is the delimiter you chose, the semi colon, followed by a press of the CTRL-O. Next, you pressed the RETURN key to include the carriage return in the search string. Then, you typed the delimiter, the semi colon, again. Next, you typed the character which is to replace all carriage returns, the at sign, and finally followed with the delimiter again.

The Wild Card Typing a CTRL-W in the search string matches any single character. Remember, however, that you must precede your CTRL-W with a CTRL-O for reasons discussed above. In other words, the string CTRL-W followed by the letters "h" and "e" between the first two delimiters will find all occurrences of the word "the" whether or not the first letter is capitalized. Never put a CTRL-W in the replacement string.

Key Popping

Each character you type gently pops your Apple's speaker. You may turn this popping off by pressing CTRL-P. Turn it back on with the CTRL-P command.

Voice the Keystrokes

ProWORDS echoes the keystrokes you type with speech. You can cancel the voicing of the keystrokes with CTRL-V. Press CTRL-V again to turn on voicing.

If you would like to turn off voicing of the keystrokes and leave it off even when you next use the program, turn off the voicing then save your defaults with the APPLE-LEFT BRACKET, save-defaults command. For complete details on using save-defaults, see "Save Defaults" in this manual.

The Sticky Space

Sometimes you want to type a word which contains a space, yet you want to insure that that entire word gets printed on a single line. In other words, you want to insure that the word will not be broken at the space. That is exactly the purpose of the sticky space character. It is used in words like Apple^_II, grade^_II., or Los^_Angeles.

The sticky space is entered by typing a CTRL-_. That is, you should hold down both CONTROL and SHIFT while you press the dash key.

Insert the Current Date or Time

Pressing APPLE-` makes ProWORDS insert the current date into your text at your cursor. Similarly, APPLE-TILDE inserts the current time. ProWORDS must, of course, already know the current date. If you have a ProDOS compatible clock installed into your Apple^_//e or if you use an IIgs, the date is already recognized; you may, however, need to set it. Setting the date is accomplished with software that was supplied with the clock card or by the control panel or the monitor on the IIgs.

By the Way

You can access the Apple's monitor by pressing APPLE-OPTION-M from ProWORDS or by typing call -151 from BASIC. You'll know you're there by the ASTERISK prompt. The monitor is the most low-level program in your computer. As its name indicates, it monitors the rest of the system. It also contains many useful commands. One of the most useful, to the //gs user, is the command to set the time and date. The command that sets the date to June 1, 1988 and the time to 10:05 is

```
=T=6/1/88 10:05:00
```

To quit the monitor and return to either BASIC or ProWORDS, type a "Q" and press return. Apple //e and //c users should type "3D0G" followed by a press of RETURN to get back to BASIC or ProWORDS.

If you don't have a clock, you can set the date and time when you first boot the disk. It is necessary to do this each time you boot your computer.

Quit

When you finish with the word processor, press APPLE-Q to quit. Before you do, however, make sure you've saved any changes to the file you are editing.

Once you type APPLE-Q, ProWORDS makes sure this is what you really want by asking, "Quit?". If quitting is what you really want, type "Y" for "YES." When you do, ProWORDS ends and returns you to the system.

Note that APPLE-Q is not the only way to leave ProWORDS and start another program. Besides using APPLE-Q to access the ProDOS quit code, you can use either the execute-program (APPLE-E) command and directly type the name of the next program to run or you can use ProWORDS' selector to move the cursor to the file to run, then press RETURN. Press APPLE-QUESTION MARK to access the selector. See "Appendix C" for complete details on the selector's operation and commands.

When the normal quit code starts, it asks you to enter the prefix to use next. The prefix is the disk or directory name with which you wish to work. After you enter the prefix, ProDOS asks for the name of the program to use next. The program must be a type "SYS" file. WORDS, TERM, FILER, CONVERT, BASIC.SYSTEM, BRL, SPELL, BAS, and EDY are all type SYS files.

The normal quit code that asks you for the prefix and name of the next application is not the only way to move from program to program. There are several excellent "selector" or "shell" programs that make the process of moving from file to file or performing disk maintenance very convenient. One of the best is a package called DAVEX. A limited version of Davex is licensed for use with the MicroTalk boot disk. For details on purchasing a complete version, type a DOLLAR SIGN at the Davex open folder prompt. When you quit any system program, you exit back to Davex where you have a variety of options including starting other programs. For more details on DAVEX, see "Appendix G" in this manual.

Grade II Braille Translation

ProWORDS lets you translate your files into grade II braille. To translate a file, press APPLE-G and type its name or point to the file with the selector and press "G".

Although you probably won't be aware of the fact, ProWORDS starts the ProBRAILLE program which, in turn, translates the specified file and reruns ProWORDS. When the translation is complete ProWORDS loads the translated file and you are ready to print it with the APPLE-P command.

Note that you should save any unsaved work before you use the APPLE-G command. The command asks for the name of the file to translate. If you have been editing the file you want to translate, the APPLE-G command types out the name of the current file for you, but, again, this does not necessarily mean that the information in your buffer and the information in the disk file are exactly the same. Save your work before translating it.

When ProBRAILLE translates a file, it assigns the new, translated version of your file a name with the period character and the characters "B," "R," and "L" appended to the original name. So, if you translated a file named TROPICS, the translated file's name would be TROPICS.BRL.

If the last line of the file you are translating contains the ch (chain) directive, ProBRAILLE asks the name of the directory to store the translated files. If you use the chain directive, you should set the prefix to the directory that contains the files. See the "Directives" section of this manual for more details on the chain directive.

In order for ProWORDS to properly translate a file, you must have the ProBRAILLE program stored in the same directory as ProWORDS. On your master disk from APH, they are already in that state. If, however, you move your programs to a hard disk or RAM disk, make sure to copy the files WORDS, WORDS.MACROS, BRL, TBL2, and B.TBL2 all to the same directory.

INFORMATION AND UTILITY COMMANDS

The following section of this manual gives background information about ProWORDS' internal storage techniques and explains another group of commands which provides you with information about cursor position, ASCII codes, and remaining memory available for document use. While it is not essential to know about ProWORDS' internal storage techniques, such a discussion provides the advanced user with information that further increases the program's flexibility.

Pseudo Lines

ProWORDS stores your text as a continuous stream of characters. It stores these characters in two different buffers in memory. When you move the cursor right, ProWORDS grabs one character from the right buffer and adds it to the left buffer. This technique permits inserting text directly to the end of the left buffer with no limit on the amount being inserted.

ProWORDS keeps track of each line for display and editing purposes only. The lines you see on the screen reflect, in no way, the actual text that will be printed. The actual format of the printed text depends on the format you specify with the printing directives. Note that you may also see an exact representation of

how your text will appear when it is printed. See "Calculate Printing Position" for more information on this procedure.

Line-Oriented Material

If you load a file that was created by another word processor which actually stores its data with carriage returns at the end of each line or load a file which was downloaded from a mainframe computer which contains carriage returns at the end of each line, you may handle the material in one of several ways. If you plan to edit the material, it is probably best to use the search and replace command to replace the carriage returns with spaces. Before you do this, however, look at the material. You may need to do some creative replacing. If, for instance, the program stored each line with a carriage return at the end of it and stored two carriage returns to mark paragraph boundaries, you may wish to temporarily replace all occurrences of two carriage returns with some unique character or series of characters. Try, for instance, replacing two carriage returns with "@@@". This will be a temporary change that will permit you to safely replace all the remaining carriage returns with spaces. Once these two steps are performed, you may want to replace all occurrences of three spaces with two spaces. After this is complete, go back and replace all the "@@@"s with two carriage returns. A macro exists which does all this for you with one keystroke. It is executed by pressing OPTION-S. Read more about macros in the "Macros" section of this manual.

Once you strip off the RETURNS at the end of each line, you may extensively edit the material without limitations characteristic of line-oriented word processors. If, after editing the material, you wish to reconstruct the file so the other word processor can handle it, you may wish to print the text to a file. This procedure, discussed in the "Printing" section of this manual, pretends that the file is your paper and formats the text exactly as it would normally print. This means that each line would be followed by a carriage return. Be sure to save the file as well in case you wish to do further editing on it because the file is much easier to edit and format when it is stored with carriage returns only in places where you really want a new line.

Calculate Printing Position

The flexibility of the directives system of formatting your text does not permit "what you see is what you get" editing capabilities. If you think about it, you'll see that "what you see is what you get" may not really be what you want. Note, for instance, at the confusion that would be caused by using some of the more advanced conditional directives which allow the printing to format one way for braille and another for ink. You may, however, take a snapshot of your document and calculate the exact position of your cursor if the paper were printed with the specified directives. (See more about directives in the "Printing"

section of this manual.) Press APPLE-K to calculate your cursor position. Depending on where your cursor is, within a few seconds, you will see the 22 lines above your cursor on the screen as they would appear on the paper if you printed the file. Your current cursor position will be indicated with an "@" on the display.

The exact position of the character under the cursor appears in the upper left of the screen. The form of this information is 3 numbers separated by periods. The first number is the page number, the second is the line, and the third is the exact column where the character under your cursor appears--if your document were printed with the directives currently specified.

You may examine this display as long as you like. It reverts back to normal only when you enter your next keystroke.

If you want only the information about the current cursor position without actually showing you how the paper will look, you may save a few seconds. Press APPLE-J for just the cursor position and not the display.

Free Memory

APPLE-F displays the amount of free memory remaining for use with your document. When you come close to running out of memory, ProWORDS warns you by automatically displaying the amount of remaining memory. Each character you try to enter beyond that point is, instead of being inserted into the text, pushed onto the keystack. You should, when this happens, find a paragraph boundary at which to break the text into two smaller files. Then use the chain directive (ch) to link the two smaller files together when you print your document. See more about the directives in the "Printing" section of this manual.

Upper Case Detect

Speech synthesis users need a way of effectively proofreading material. One characteristic of particular importance is the case of the letters. You can easily tell the case of the letters you type by the raising of the pitch when that letter is typed. When you read, however, capital letters are not indicated. You can make ProWORDS raise the pitch of capital letters as you read. Do so by pressing CTRL-6.

When upper case-detect is active, each capital letter in the text--no matter the reading mode--is pronounced with an elevated pitch. You will not care to use this feature all the time, but for proofreading, it is quite useful.

The upper case-detect command is toggled on and off with the CTRL-6 command.

Toggle Case

What happens when you find entire paragraphs or even pages of material all typed in upper case? Adjusting such material to your taste could, if it were not for the toggle-case command, prove quite time consuming. The toggle-case command, as the name implies, toggles the case of any character the cursor crosses. In other words, if the character is typed in lower case, moving the cursor across it forces it to upper case; moving back across it brings it back to lower case.

To use the toggle-case command, hold down BOTH the APPLE and OPTION keys and use the RIGHT and LEFT ARROWS to move the cursor across the appropriate text.

ASCII Code

CTRL-APPLE-A displays the ASCII code of the character immediately to the cursor's left. This proves useful for determining printer control codes.

Preserving Defaults

The default conditions--marks, key clicking, voicing, increment values, etc.--may be preserved at your favorite settings so that ProWORDS uses those values every time you start the program. To preserve your current settings as defaults, press APPLE-LEFT BRACKET. ProWORDS asks, "SAVE PROWORDS DEFAULTS?" (Y/N)" If this is what you really want to do, press "Y" and the conditions currently set remain in effect every time you start the program. If you change your mind, press "N" or press ESCAPE.

Note that the default printing values may not be changed. If you think about it, you'll see that you really wouldn't want to save them. Think, for example, of sharing documents with other ProWORDS users. When everyone's program assumes the same printing values, the document prints equally well on everyone's program. Use a file of directives or a macro containing commonly used directives to change the standard values of your printing settings.

Macros

Macros are sets of keystrokes or commands that are executed with a single command. Whenever you've got something that needs doing every day, you've got a candidate for a macro. They are especially useful for repetitive typing such as commonly used phrases or formatting commands. In addition, however, macros may be used to execute a series of commands, thereby automating complex search and replace sequences or running other programs with a single keystroke.

Executing Macros Every macro command is executed with the use of the OPTION key. When you hold down OPTION in conjunction with the letter of the desired macro, the keystrokes and commands in that macro will be entered just as if you typed them from the keyboard.

Defining Macros Your disk comes with several useful macros already defined. You may, however, define as many additional macros as there are keys left on the keyboard. Macros are automatically loaded when ProWORDS is started. They come from a file on your disk called WORDS.MACROS. WORDS.MACROS is a standard text file so you can load it, edit it, and then save it back to disk as you think of additional useful functions for macros. Don't forget, though, that if you add macros to the macro file, you should execute ProWORDS again to force it to use the new macros.

If you move ProWORDS to a different directory, be sure to move WORDS.MACROS, too. ProWORDS loads the file WORDS.MACROS from the same directory where the program itself is located.

Structure of the Macro File A macro command file, like many other kinds of files, possesses a special structure. The first thing you should know about every macro file is that it must start with a CTRL-A. ProWORDS uses the CTRL-A to verify the file as a macro file. You don't need a CTRL-A between every macro; just one at the beginning of the entire file for verification. Remember, if you want to type CTRL-A, or any other control character, use the CTRL-O (override) command to tell ProWORDS that the following character is to be placed into the text and not acted upon as if it were a command.

After the CTRL-A, the macro file contains several consecutive entries. Each entry is a macro.

The Parts of a Macro Every Macro contains four parts. They are (1) the trigger character, (2) the initial delimiter, (3) the macro definition, and (4) the final delimiter. The next macro's trigger character follows the final delimiter of the previous macro.

Trigger Character A macro entry in the file always begins with the character that will be used (in conjunction with the OPTION key) to execute the macro. This character is called the trigger character because it triggers the macro into action. If you want to make a macro that makes OPTION-R type your return address, the trigger character would be "R."

Initial Delimiter After the command character for each macro entry in a macro file comes a delimiter. The delimiter can be any character that you won't be using in your macro. It is used to signal the end of this macro. In other words, ProWORDS notes this character and executes everything in your macro definition until the next character that matches your initial delimiter. This

permits you to use any character you want in your macro including RETURN.

A common delimiter to use is the "@" character. "@" is used for several reasons. First, it is seldom used. If you find you need a macro that contains the "@" character, just select a different character as your delimiter for that macro. "@" is also used because it makes it easy to find a trigger character in the macro file. If you wanted to find the macro that defines OPTION-R, you could find "@R@". That is the final delimiter from the previous macro, the trigger character of interest, and the initial delimiter of this macro. Of course, this scheme of finding individual trigger characters doesn't depend as much on the character being "@" as it does that all or most macros use the same delimiter.

The Macro definition Once you decide on a command character and a delimiter, the rest is easy. Just type the characters to include in your macro. To include them in the macro, just type them as if you were normally typing them. If you wanted your macro to type "today is a nice day", you would simply type "today is a nice day" as the definition part of your macro.

The Final Delimiter When you finish entering and editing your new macro, signal its end by typing the delimiter character again. The next macro's command character immediately follows the previous macro's final delimiter character. Don't be tempted to add extra RETURN characters between macro definitions.

Macro Example Following is an example of a macro that types APH's address. The trigger character is "R". The delimiters are "@" signs. Feel free to edit it for your own address.

```
R@MicroTalk
1839 Frankfort Ave.
Louisville, KY 40206
@
```

Notice the RETURN characters used in this macro. To represent them as they appear on the screen, they are denoted in this example by moving down to the next line. If they were represented as ^M, the macro would look like this:

```
R@American Printing House for the Blind^M1839 Frankfort
Ave.^MLouisville, KY 40206^M@
```

Don't forget, RETURN is just another character to the computer; as long as it isn't the delimiter, you can use it freely in your macros.

Entering Apple Commands into your Macros Since the APPLE and OPTION keys have no actual ASCII code, it is necessary to work out a system to represent them in ASCII text files. The solution is to

use CTRL-O to precede one of three numbers, 0, 1, or 2. CTRL-O followed by 0 stands for the APPLE key. CTRL-O followed by 1 means the OPTION key, and CTRL-O 2 indicates that both the APPLE and OPTION key are to be considered as down.

Don't forget that in order to embed a CTRL-O into your text, you must precede it with a press of the CTRL-O. (Remember, the first CTRL-O allows you to enter any control character into your text, the second one is the character to embed.)

After the CTRL-O and the number, press the key that is used to actually execute the command. If you wanted to execute another program, you would use the letter "E" in conjunction with CTRL-O and "0" because "0" indicates the use of the APPLE key and "E" is the key to use with the APPLE key. If, for instance, you are defining a macro that tells OPTION-T to execute ProTERM, the macro would look like the following:

```
T^00E/user/util/term
@
```

The macro's trigger character is T. That's why it comes before the first delimiter.

The first delimiter of this macro is "@". After the first delimiter, the macro contains a CTRL-O. (That is what is represented with "^O".) Remember, CTRL-O is used to indicate that an APPLE/OPTION command is coming. The number immediately following the CTRL-O is 0. That means the APPLE/OPTION command is one that involves the APPLE key only. The character following the CTRL-O and the "0" is "E". Those three characters tell the program to make an APPLE-E. If you press APPLE-E from the keyboard, ProWORDS says, "execute" and asks you for the name of the program to use next. You would type the name of that program and press RETURN. The macro acts the same way; once it presses the APPLE-E (with CTRL-O0E, it should type the program's name and RETURN just as if you were typing it from the keyboard.

The name of the program and its final RETURN are followed by the terminating delimiter, the "@". (Again, that's why the at sign appears on the line by itself in this manual.) The next macro definition character would follow this terminating delimiter immediately.

There are several macros already setup on your disk. They are as follows. Remember, each macro is issued by holding down the OPTION key with the indicated character.

Table of Supplied Macros

B sets up the SlotBuster multi-function card to use the parallel port.

I sets the SlotBuster to use the serial port.
C converts AppleWorks control commands to ProWORDS control commands.
F opens the /user/util folder then executes FILER.
S strips carriage returns from files that contain a hard carriage return at the end of every line. This macro is especially for processing text that you download from bulletin boards.
Q quits and loads the menu program.
RETURN types my address and the current date for use at the top of letters. You can edit the macro to type your address.
DASH types the begin underlining directive.
EQUALS types the finish underlining directive.

Adding to the Supplied Macros Chances are that you'd like to keep the supplied macros and add to them. If you want to do that or if you want to add to your own macro file, follow these steps.

1. Load the file WORDS.MACROS from the folder where ProWORDS is located.
2. Make sure there isn't already a macro with the trigger character you want to use. Do this by using the find command to find "@", the letter of interest, and another "@". If you get the "Not Found" message, you're ready to proceed to step 3. If you do find the command, decide if you want to keep it or not. If you do, select another trigger character and repeat step 2. If you don't want to keep the old macro, delete it. This is most easily accomplished by moving the cursor to the right of the first delimiter, setting mark one, then using the find command to find the second delimiter. (End the find command with DOWN ARROW to indicate the find should go from your cursor down.) When the find command finds the second delimiter, press OPTION-DELETE to delete from your cursor back to where you set mark one. The cursor is now positioned to add the new macro definition. At this point, you might want to press RETURN a couple times just to make working with the new macro clearer. Don't forget, though, to delete those RETURNS before you save the macro file. That is, type the trigger character, the first delimiter, the macro definition, and the final delimiter.
3. When you finish editing the macro file, save it. You'll need to execute ProWORDS again before you can test the new macros. There is a macro supplied with the disk that does this for you. It is accessed with OPTION-TAB.

DISKS AND FILES

The following group of commands lets you work with files on the disk. A file is a group of information with a name. You give the text you write a name when you save it to disk. You use that name later when you want to work with the text again.

In each of the following commands, ProWORDS asks you to supply a name for the file. It displays a prompt that says, "TYPE FILE NAME. If you have changed your mind when you see this prompt, press ESCAPE to cancel the function and return to your work. If you don't know the name of the file you want to work with, press QUESTION MARK to see a list of files. Once you see the list, use the arrow keys to move to the file you want, then press the letter of the function to operate on that file. Read more about this "selector" in "Appendix C."

The table that follows shows all the disk commands available from ProWORDS. It shows the key to press while holding down the APPLE key and the function that command performs.

Table of Disk Commands

A appends to a file.
B saves a block of text to disk.
C copies a file.
D deletes a file.
E executes another program.
G translates a file into grade 2 braille.
I inserts a file.
J calculates printing position.
K calculates position and shows how printing looks.
L loads a file.
M makes a new folder.
N changes the name of a file, folder, or volume.
O allows you to load material that did not fit into memory.
P prints the file and any chained files.
Q quits the program.
R receives data from an external source.
CTRL-R resets I/O port before receiving data.
S saves the file.
T transmits a file via one of your serial ports.
CTRL-T scans for an I/O port then transfers data.
U unlocks or locks a file.
V shows you the contents of a file.
COMMA sets the prefix.
QUESTION MARK shows a list of files in the current folder.
SLASH shows a list of files in the main folder.
PERIOD displays all disks in your drives.
LEFT BRACKET saves ProWORDS defaults.

The following paragraphs describe the commands specific to ProWORDS. There are several more disk commands which are common to both ProWORDS and ProTERM discussed in "Appendix C." These include useful functions like copying files, deleting files, locking files, viewing the contents of other files, and seeing what files appear on any disk. It is important to read "Appendix C," so don't forget.

Saving Your Text

In order to keep a permanent copy of what you have written, you must save the text to a disk file. Saving your work to a file on disk requires two steps. First you must use the save (APPLE-S) command. When you press APPLE-S, ProWORDS puts a message on the bottom of the screen. This message depends on what you have already done. If you haven't loaded a file from disk or if you haven't already saved the text, the message will look like this:

SAVE

Press "?" for a list of files or ESC to return to editing or

TYPE FILE NAME

If you already loaded a file from the disk or if you already saved the text, the message displayed depends on the name of the file that you loaded or saved. If, for instance, you loaded a file named "test", the message would look like this:

SAVE

Press "?" for a list of files or ESC to return to editing or

RETURN to accept the displayed name or

TYPE FILE NAME

Assume, for a moment, that you haven't yet loaded or saved a file. The second step in saving your work is to give this text a name. Not just any name will do however. You must obey the ProDOS file name rules. You may read more about these rules in "Appendix C" of this manual. In short, they state that a file name must start with a letter and may contain no punctuation except for the period (.). You don't have to worry too terribly much about this rule because ProWORDS won't let you type an invalid name. You might wish to be aware, however, of this fact when you hear the computer buzz as you enter an illegal character.

When you decide on a file name, just type it in.

If ProWORDS displays the name of the last file you loaded or saved and you want to use a different name, just begin typing the new file name. If, on the other hand, you want to use the name displayed, just press RETURN. If, instead, you want to see a list of files already on the disk, press QUESTION MARK and use the arrows to point to the file of interest.

When you enter the file name and press RETURN, the disk spins as your file is stored. When the disk quits spinning, the "SAVE TYPE FILE NAME" message disappears and you can continue editing.

The next time you wish to save this file, which should be every 15 or 20 minutes, and you press APPLE-S, ProWORDS automatically types the name of the last file you used. Remember, press RETURN to use that name or type in another name to save the text to another file.

Block-Save

Rather than saving the entire contents of your workspace, you can save just a block with APPLE-B. This multiple-step procedure is described below:

1. Move your cursor to the top of the segment of text to save.
2. Set mark one with OPTION-1.
3. Move your cursor to the bottom of the segment of text to save.
4. Press APPLE-B.
5. When ProWORDS asks for a name for this segment of text, type a file name and press RETURN.

If you change your mind about saving this segment of text, press ESCAPE to return to editing.

If the file you specify with the block-save command already exists, ProWORDS announces, "ADDING TO END OF EXISTING FILE" and appends the marked block of text to that file. Note that this permits taking notes on material you read and storing all the notes to another file.

The block-save command proves especially useful when you are writing or reading text and need to save a portion of that text to another file which you continually add to. If, for instance, you are writing and receive a phone call, you may want to set mark one (OPTION-1). After setting mark one, simply type the information to note. When you have typed the relevant information, use the APPLE-B command and add that information to the already existing file of notes. Then, when the block is saved, use OPTION-DELETE to remove the information from your current work and resume where you left off when the phone rang.

Loading Files

Load a file from disk into memory with APPLE-L. ProWORDS loads the entire file (or as much of it as fits) into memory for editing. If the file is too large to fit into memory all at once, you have some options on what to do. See "Overflow-load" for more information.

Normally, ProWORDS expects you to load files that you created with ProWORDS. These files have a standard format and are assigned a file type that you can see by using the selector. ProWORDS also loads files that were created with other programs. ProWORDS loads any file with the file type "TXT". It also works with files created by the AppleWorks (TM) word processor. AppleWorks files use a file type of AWP. AppleWorks files are so different from standard TXT files that ProWORDS actually translates the AppleWorks file into a TXT file then loads that TXT file. ProWORDS gives the TXT file the same name as the AppleWorks file and adds a PERIOD (.) and the letter "A" to the name. If the file is extremely long and you don't have enough memory in your computer to hold it all, ProWORDS breaks the file into several sections giving each section a name with another letter of the alphabet.

When you load a file from disk and make changes in it, you must save it back to the disk to make those changes permanent. You are only working with a copy of the file; the original on the disk is unaffected until you overwrite it with the save command.

When you use APPLE-L to load a file after having already loaded or saved a file, ProWORDS will ask you if you want to use the current file name again by typing it out for you. If you wish to use that name, just press RETURN. If you want to load a different file, just type the name of that file.

When a new file is loaded, whatever was previously in memory disappears; so, be sure to save your work to disk if you plan to use it again.

Overflow-load

If you try to load a file that won't completely fit into the available memory, ProWORDS displays the "FILE TOO LARGE" error message. If this situation occurs, you may still access the rest of that file. Depending on what computer you use, there are several ways to go. If you are using an Apple //gs, and you have set up a RAM disk, you may wish to consider making the size of the RAM disk smaller, giving more RAM to your application, ProWORDS in this case. Remember that if you change the size of your RAM disk, it is necessary to "cold start" your computer. You can do this either by turning off the machine, waiting for about 30 seconds, then turning it back on, or you can use the "boot -i" command from DAVEX.

If you don't have a //gs or if you don't have enough RAM to load large files, you can load sections of the file. When you see the "File too Large" message, save the portion of text you did manage to load to a new file and then use the overflow-load command (APPLE-O). This command begins loading the file from where the last loading command stopped. If following attempts still produce

the "FILE TOO LARGE" error message, repeat the procedure as many times as necessary.

It is actually not necessary to immediately save the first chunk of the file. It is perfectly alright to edit, reformat, or change the file in any way before saving that file with another name. You may, in fact, wish to continue loading the large file without saving the first sections. Remember however, that if you want to access this file frequently, it is better to break it down into manageable chunks.

Insert a File

ProWORDS allows you to insert the contents of another file at your cursor. The command to invoke this feature is APPLE-I. When you press APPLE-I, ProWORDS asks you to type the name of the file to insert. Type the file name or the pathname and press RETURN. When the file is loaded, ProWORDS returns to editing with your cursor in the exact place you left it and with the contents of that new file immediately to the cursor's right.

Appending Files

You can add the text on which you are working to the end of an existing file on disk. Use APPLE-A to do this. When you press APPLE-A, ProWORDS asks for the name of the file to which you wish to append the information in your workspace.

If you want to add only a portion of your workspace text to another file on disk, use the block-save command discussed in the "Block-save" section of this manual.

Receiving Information from Other Devices

You can receive text from other computers via one of your Apple's serial ports. To receive a file, press APPLE-R.

When you press APPLE-R, ProWORDS looks at how many serial ports your computer has. If you have only one, it immediately asks for the name of the file where the information you are about to receive is to be stored. Type the name of the file and press RETURN.

If your computer contains two or more serial ports, ProWORDS asks you which one to use. The next time you use either the receive or transfer command, ProWORDS assumes that you wish to continue using the same serial port. If you decide that you want to use another port, press CTRL-APPLE-R to make the program ask for the port to use again.

When you type the name of the file and press RETURN, ProWORDS tells you to begin the transfer and to press ESCAPE when the transfer is

complete. This process does not effect the text in memory nor does it effect the cursor's position.

In order to receive long files, your computer and the other device must agree on how they will control the flow of information. (If, for instance, the text fills the available memory, the Apple must tell the other device to pause while the material is stored to disk.) This process of communication is called handshaking. There are several kinds of handshaking available. Each, of course, has its own name. Sometimes, the computer uses voltage levels on the wires in the cables to see if the other machine is ready to either send or receive data. This is called "hardware" handshaking. Hardware handshaking is often used when the two machines are in close proximity. Another form of handshaking is software. Software handshaking does not depend on the voltage levels of any of the wires in the cable to determine the other machines state of readiness. Instead, the software method of handshaking sends a special character in the stream of text to tell the other computer to pause sending while the receiving machine takes care of chores. This method of communicating has come to be called X-ON X-OFF protocol. The X-ON character, the CTRL-Q, turns on the flow of information and the X-OFF character, the CTRL-S, turns off the flow of information until it receives the next X-ON character.

If you want to use software handshaking (the preferred method on the Apple with its built-in ports), you should tell your Apple's serial port to do so. If you're using an Apple IIgs, you can permanently set handshaking to software by using the control panel. If you're using a //e or //c, you will need to use a "serial port initialization string." A serial port initialization string is a series of special characters that give commands to the serial port to set certain attributes. You may, for instance, want to use such a string to set the port to 1200 baud, 8 data bits, 1 stop bit, and software handshaking. Apple //e users can use the switch settings on the Super Serial Card to set most of the common settings to normal values so you don't have to use a serial port initialization string to set them. The software handshaking is not, however, one of the options available with the DIP switches on the SSC.

Each serial port initialization string begins with a certain character that lets the port know that a command is coming. Usually, this character is either a CTRL-a or a CTRL-I depending on whether your port is set for printer or modem mode.

After the port initialization character, there are more characters that describe the command to perform. These next characters take one of two forms; either two letters or a number and a letter.

The commands using two letters either enable or disable a function of the port. They use the first letter as an indicator of the function. The second letter, either a "D" or a "E" indicates either "disable" or "enable."

The command, for instance, to enable X-ON X-OFF handshaking is

<CTRL-I>X E

This is the command character, assumed to be CTRL-I, followed by the letter "X," a SPACE, and the letter "E". This is easily remembered by noting that the "X" stands for "X-ON X-OFF" handshaking and the "E" stands for "enabled." To disable X-ON X-OFF handshaking, you would use the command:

<CTRL-I>X D

The SPACE between the "X" and the "D" or "E" is essential.

The second form of command consists of a number followed by a letter indicating the function to adjust. The number is used to specify the value of the adjustment. You might set your port's baud rate to 9600 by using the command

<CTRL-I>14B

The following chart shows the values of numbers and the meanings of the letters used in serial port initialization strings. Remember, if your port is set to modem mode, the command character is CTRL-A, if it is set to printer, the command character is CTRL-I.

Table of Serial Commands The first group of commands are followed by the SPACE character and either "E" to enable the function or "D" to disable it. The command letter and function are shown.

L Suppress adding linefeeds to carriage returns.

M mask incoming linefeeds.

X X-ON X-OFF handshaking.

The following commands each use a number and a letter. First, the letter is shown followed by the number corresponding to each value.

C Carriage return delay

0 = no delay.

1 = 32 milliseconds.

2 = 250 milliseconds.

3 = 2 seconds.

D Data Format

0 = 8 data bits, 1 stop bit.

1 = 7 data bits, one stop bit.

2 = 6 data bits, 1 stop bit.

3 = 5 data bits, 1 stop bit.

4 = 8 data bits, 2 stop bits.

5 = 7 data bits, 2 stop bits.

6 = 6 data bits, 2 stop bits.
7 = 5 data bits, two stop bits.

Permanent File
Research Library
A.11

F Formfeed Delay

0 = no delay.
1 = 32 milliseconds.
2 = 250 milliseconds.
3 = 2 seconds.

L Linefeed Delay

0 = no delay.
1 = 32 milliseconds.
2 = 250 milliseconds.
3 = 2 seconds.

P Parity

0,2,4, and 6 = no parity.
1 = odd parity.
3 = even parity.
5 = mark parity.
7 = space parity.

B Baud Rate

0 = use switch settings.
1 = 50.
2 = 75.
3 = 110.
4 = 135.
5 = 150.
6 = 300.
7 = 600.
8 = 1200.
9 = 1800.
10 = 2400.
11 = 3600.
12 = 4800.
13 = 7200.
14 = 9600.
15 = 19,200.

Usually, you'll want to set the DIP switches or the control panel to set the default values of your serial port to values that you most commonly use then use the initialization strings to alter these settings when using another device. The X-ON X-OFF setting, however, is not adjustable via the DIP switches, so you will have to set that one. ProWORDS comes configured to use this string already.

Apple //c users should note that the built in serial port's default is to use two stop bits. Keep this fact in mind when trying to communicate with other devices. Either set the other device to use two stop bits or use the D command with the appropriate value to change the data/stop bit configuration to the requirements of the other device. The command <CTRL-I> OD sets the serial port to use 8 data bits and 1 stop bit.

Transmitting Text to Other Devices

In addition to receiving text from other computers or devices, ProWORDS lets you transmit files to other devices via one of your Apple's serial ports. Press APPLE-T to initiate a transfer.

When you press APPLE-T, ProWORDS checks to see how many serial ports you have. If you have only one, ProWORDS asks for the name of the file to transmit. If, on the other hand, you have more than one serial port, you are requested to specify the one to use. Once you've specified this information, subsequent transmissions are assumed to use the same port. If another port is desired, press CTRL-APPLE-T to initiate the transmission and prompt for the port to use.

PRINTING

The APPLE-P command processes your text for printing. When you use the APPLE-P command, ProWORDS interprets directives which you type into your text to produce a document that is both well formatted and infinitely adjustable.

What are Directives?

When you print, ProWORDS assumes certain things about your printing job. It assumes, for example, that your printer is connected to slot 1, the standard place for printers. If your printer is connected to a different slot, you use a directive to tell ProWORDS to change that assumption. These "assumptions" are called defaults. You can see a complete list of ProWORDS assumptions (or defaults) about your printer in the table that follows.

Directives are a special kind of text that allow you to change the assumptions ProWORDS makes about your documents. Directives do not get printed; they only control the printout. You might wonder how the computer distinguishes between what you want to type and what you want to be a formatting command. Actually, it is quite simple.

The Format of a Directive

Directives have a specified format that is used to distinguish them from normal, printing text. There are two kinds of directives, the print directive and the inclusive directive. Each has a special

format. Each contains a beginning character or sequence of characters, each contains a two-letter mnemonic, each may optionally contain parameters, and each ends with a terminating character. The difference between the normal directive and the inclusive involves the function that is required. Inclusive directives may appear anywhere within a line. They always begin with the CTRL-\ character and end with a CTRL-]. Inclusive directives are used for special functions like underlining, italics, or tabbing to the next tabstop; functions, in short, that must occur within a line.

Normal directives always begin with the period character. Obviously, you don't want every period you type to be interpreted as a directive. Therefore, an additional restriction is applied to qualify text as a normal directive; the period must either be the very first character of the file or it must immediately follow a carriage return. You will notice that this distinction effectively renders normal directives adjustable only at line or paragraph boundaries (after all, a carriage return must precede the command character). It so happens that all the normal directives do not require adjustment within a line. For printing specifics that require interline formatting control, the inclusive directive is provided. Obviously, the advantage of the normal directive is its ease of use and reading. You may, however, significantly increase the ease of use of inclusive directives by referring to the "Macros" section of this manual.

As with inclusive directives, normal directives use a terminating character. Normal directives use the carriage return to terminate the directive. This makes normal directives very easy to read and edit. Do not worry about printing an extra line on your document with these directives; the terminating carriage return is used only to end the directive. It will not be printed.

Inclusive directives use CTRL-RIGHT BRACKET as the terminating character.

Changing the Format of your Document

You can use directives to alter not only the assumed values--left margin 12, right margin 75, top and bottom margins of 6--but you can use them to alter the format of your document at any point. New directives take effect at the point in your text where they appear. This proves especially useful for indenting blocks of text or "undenting" blocks of text. (Undenting refers to the formatting procedure where the first line of a paragraph actually appears left of the rest of the block of text.)

The directive technique is especially useful as both a powerful formatting language and as a permanent record of your preferred specifications.

You might create files or macros of your favorite formatting directives then, when you finish a paper, insert the file or press a macro key to insert the proper keystrokes in the appropriate place in the document.

Remember that a print directive takes affect only when it is reached. In other words, if you set the top margin to eight on the middle of the first page, obviously only the second and subsequent pages will have a top margin of eight lines. The first page will have, if no previous top margin directive (tm) had been used, six lines. Recognize that this allows your work to contain multiple formats within a single document. Note too, that you don't need to use a directive unless you want to alter that aspect of the format. Directives, in other words, are used only to change the assumed or current values of formatting specifications.

Initiating and Controlling the Printout

When you press APPLE-P, ProWORDS asks you to either press RETURN for a single copy or to enter the number of copies desired. (You may also press ESCAPE to return to editing.) When you either press RETURN or type a number and then press RETURN, ProWORDS begins formatting your text as specified by either the default values or the directives. The assumed values of each of the directives is described in Table 5-1.

If you type a number in response to the request for the number of copies to print, ProWORDS prints your entire paper that number of times. These multiple copies are printed consecutively; in other words, your complete document is printed, then the complete document is printed again. If you have used the ch directive to chain a number of files together, ProWORDS loads the original file from the disk and resumes the printing with the next copy. You can get complete details about the ch directive in the "Directives" section of this manual.

If you want to print several copies of a file and want to know where each copy begins, use the ps, pause, directive at the beginning of the document. This directive makes you press space to continue.

Pausing a Printout

As ProWORDS prints your document, you may pause the printing with a press of the SPACE BAR. Resume printing with another tap of the SPACE BAR.

Cancelling a Printout

If you see a mistake in your printout, you have a choice about how to interrupt the job. You may either press the ESCAPE key, in which case your cursor is returned to its original position, or you

can press RETURN. If you press RETURN, your cursor is placed at the point in the printout where you discovered the error.

Notes About Buffers

If you are using a printer with a large buffer or if you are using a printer buffer, the technique for pausing and cancelling does not appear to work. It works, but the text that has been sent to the printer is far beyond the text that is actually printing. (This is the purpose of printer buffers.) Printer buffers store as much text as possible and feed it to the printer as it requires. To terminate a printing job when using a printer buffer, it is necessary to dump the buffer. You will need to consult the buffer's manual to determine the exact means of doing this.

The following table of print directives outlines the function and parameters of each directive. Each entry notes common uses, default settings, and parameter ranges. Remember that each directive is preceded with the period command character.

Normal Directives

Function	Code	Default	Description
----------	------	---------	-------------

Printer slot:	pr 1		The pr directive tells ProWORDS where your printer is located. If it is in slot 1, you don't need to use this directive unless you are performing one of the special functions associated with the pr directive or if you have more than one printer connected to the computer.
---------------	------	--	---

There are seven slots in your Apple (unless you are using a //c in which case you effectively have two slots- slot one for the printer and/or the modem and slot two for the synthesizer.) If you need to use a printer in a different slot, type the number with the pr directive that corresponds to the slot in which your printer is connected. A printer in slot two, for instance, would be accessed with the directive pr2. The pr directive should be the first item in your text.

Slot number 0 corresponds to the screen and the speech synthesizer. You may use pr0 to proof the formatting of a document. If you see an error in formatting, press RETURN; your cursor will be placed at the point in the document where you discovered the error. If you press ESCAPE, your cursor is left in its original position.

Besides the normal seven slots in your Apple, ProWORDS allows you to use two additional "token" slots. Tokens are values that don't necessarily relate to a specific slot but, rather, stand for an instruction. The token values you may use with the pr directive are 8 and 9.

If you use 8 as a parameter with the pr directive, an exact duplicate of what appears on the paper will be written to a file on

your disk. ProWORDS always uses a file named "P" to write the "print" file. Something else you should know about the way the pr8 directive works is that it always adds to the end of the file if there is already a file named "P" on your disk. You should, therefore, delete any files named "P" before you use the pr8 directive. Another thing to know about the pr8 directive and the P file is that the P file is always created or added to in the currently open folder.

If you use "9" as the token for the printer slot, ProWORDS will print to nowhere. A valid question for using such a parameter might be, "why would I want to print nowhere?" There are several reasons for printing nowhere. See the sections "Calculating Current Position" and "Generating the Table of Contents".

Left Margin: lm 12 The left margin directive allows you to alter the number of spaces skipped before text is actually printed on a line. When setting up your printer, you can always align the printer margins with the computer by using lm0 as the left margin then printing. If the first character on the line is exactly aligned with the paper's left edge, your printer's left margin is correct.

Absolute vs Relative. All margin directives take either absolute or relative values. An absolute value is specified when you type the value directly. If, for instance, you used the directive lm17, you are telling the program to begin printing each line 17 spaces from the left edge of the paper. The relative value, on the other hand, is used to adjust the margin. A directive like lm+5, for instance, makes each line's left margin move in five spaces from whatever it was. Relative parameters may be either positive, as the example shows, or they may take negative values. The directive lm-5 moves the left margin five spaces left of where it was.

These relative values permit numerous variations on the way your text is formatted. If you need to indent a block of text such as a quotation, use the lm+5 and rm-5 directives to set the block of text apart from the main body. (This sequence of directives makes both the left and right margin five spaces larger than they were.) At the end of the block of text, use lm-5 and rm+5 to reset the margins to their former values.

Right Margin: rm 75 The parameter to the right margin directive refers to the column past which no character will print. If, for example, you have the rm directive set to 75, the default, no character on the paper will print past the 75th column on the page. The words that would ordinarily extend past this limit will, instead, print on the next line on the page. You may have to adjust the rm value according to the size of typeface you are using. If, for instance, you use a printer which uses pica rather than elite typeface, you may wish to extend the right margin far beyond the normal pica value of 75.

Top Margin: tm 6 The tm directive, like its counterpart the bm directive, specifies the number of blank lines to leave at the top of each page before normal printing resumes. Note that titles may appear within the margins of each page. (See "Titles" for more details on headers, footers, and titles.)

When first setting up your printer or when trying to align the paper for proper printouts, use the tm0 directive followed by a line of text. When you print the document, the line of text should appear on the extreme top of the page. If it doesn't, make sure you align the paper lower in relation to the printer's printing head.

Bottom Margin: bm 6 The bm directive specifies the number of blank lines to leave at the bottom of each page of your printout. As with the tm directive, title lines may appear within the lines of a bottom margin.

Page Length: pl 66 The pl directive specifies the total number of lines (including top and bottom margins) that are printed per page. If you don't alter the default top and bottom margin values, the default page will print 54 lines of text. There will be 66 lines on the page. Six of them are devoted to the top margin, 54 are printed with text, and six are reserved for the bottom margin. The default page length when using the pb (print braille) directive is 25 lines. If you use a printer like the MBOSS 1 from VTEK, you will need to change the page length to 27.

Page Number: pg0 The pg directive is used to set the page number to a different value. ProWORDS keeps track of the number of pages printed even if you don't choose to put that number on the pages. You use the pg directive to change the value of the current page number. It can be useful, for instance, when printing several pages of preliminary material. When you are actually ready to begin numbering the pages with the number 1, use the directive pg1.

Spaces to Skip: sp0 The sp directive specifies the number of blank lines to skip between printing lines. To make your paper print with double-spacing, use sp1. To leave two blank lines between every printing line (triple spacing), use sp2.

Paragraph Indent: pp 0 Specifies the number of spaces to indent paragraphs. While 0 is the default value (no indenting), you may set the paragraph indentation to either a positive value (to indent paragraphs) or a negative value (to undent paragraphs). Specify the direction of indentation (either positive or negative) with the + or - characters preceding the actual number. To indent every following paragraph five spaces (until instructed otherwise), use the directive pp+5. (Actually, you may just as effectively use pp5.) To undent paragraphs five spaces, use pp-5. Note that if you want to format a block of text so that the first line extends

five spaces left of the remaining lines, you could use a sequence of directives like this:

```
.lm+5 (scoots the left margin five spaces right of where it is.)  
.pp-5 (specifies that every line after a carriage return should be  
pulled five spaces left of the left margin.)
```

Using the NUMBER SIGN character as the number to indent or outdent provides ProWORDS a way to format your work correctly for both print and braille. The NUMBER SIGN is a token value that tells ProWORDS to use the value 5 for ink printing and the value 2 for braille. It is, therefore, much better to use the NUMBER SIGN token than to use fixed values if you plan to produce documents in both ink and braille.

Titles, Headers, and Footers: `ti 1-7,0` The `ti` directive specifies one of seven different title lines that can appear on each page of your text. Title directives take two parameters; the number of the title and the line number on which to print the title. Each of these parameters is separated with a comma.

In addition to specifying the title number and the line on which to print the title, you must specify the contents of the title. The title contents consists of three parts. One section is left justified on the line, one is center justified, and the other is right justified. Each section of the title is separated with a delimiting character. (The delimiting character must be a character that does not appear in the text of the title.)

An example of a title directive that prints "Chapter 1" center justified on the third line of every page would look like this:

```
.ti1,3//Chapter One// .
```

Notice that this title was assigned the number one when the `ti` line was typed. There may be up to seven different titles. You can use as many or as few as needed. To cancel a title line, define it to print on line 0. To cancel the line in the example, the directive would be `ti1,0`; this tells ProWORDS that title number 1 is no longer active.

You may change titles at any time. To change the contents of a title, simply use another `ti` directive with the same title number as the title to change.

Default Title Margins: `dt0` The `dt` directive lets you decide how to align your titles. If you don't use the `dt` directive or if you use `dt0`, your titles are aligned with the margins in use for each page where the title appears. The title aligns itself, in short, with the text on that page. In some cases, you'll want to keep the titles aligned with the actual page rather than the text on that page.

The dt directive lets you do just that. The dt directive takes 5 parameters. First, use 0 or 1 to turn off or on the use of default page values for titles. Next, give the dt directive the left and right margins for ink and braille in that order. The dt directive that sets up default title printing for a page of 0 and 80 for ink and 0 and 40 for braille looks like this: dt1,0,80,0,40.

Odd Titles: ot0 The ot directive lets you refine the way titles are printed. If you use ot1, only the even numbered titles print on even numbered pages and odd numbered titles print on odd numbered pages.

Page Numbering: You may tell ProWORDS to print the page number on each line of your work. You do so by including a "token" character in the title. A token character is a symbol that is typed but that will not get printed. Instead, the token is replaced, in this case, with the actual value of the current page.

The token character to put into a title directive is the DELETE character. Note that the DELETE character is different from the DELETE key command. Both are accessed with the DELETE key; the difference is that if you want ProWORDS to put the DELETE character into your text rather than deleting the character left of the cursor, as a press of the DELETE usually does, you must precede the press of DELETE with a press of CTRL-O. CTRL-O is the override command. It allows you to override any command that a key might perform and insert that keystroke directly into your text.

Following is an example of a title line that causes the current page number to be printed in the center of the 63rd line of each page of your paper. Note that the characters represent the DELETE character.

```
.ti1,63//<DEL>//
```

You can make hyphens flank the page number with a directive like this.

```
.ti1,63//-<DEL>-//
```

Number Braille Pages: nb 1,0 Printouts in braille usually don't include separate lines just for the page number. Instead, braille pages are usually numbered so that the page number prints at the right end of the first line of text. It is for this reason that if you use the pb (print in braille) directive, ProWORDS automatically numbers your pages as described. If you want a different page numbering format, or if you don't want page numbers printed at all, use the nb (number in braille) directive to change these characteristics.

The nb directive takes two parameters, each separated with a comma. The two parameters are each a number. The first is used to

describe the line number on which the page number will be printed. Using zero as the first number turns off printing of braille page numbers. The second parameter describes the method of numbering pages. If the second number is an odd number, ProWORDS prints the page number only on odd numbered pages. This is useful when printing on printers that support interpoint. (Interpoint is a type of braille printing where dot patterns are embossed on both sides of the paper.) If the second number in the nb directive is an even number, page numbers are printed only on evenly numbered pages. If the second number is zero or if it is omitted, page numbers get printed on every page.

See more about printing in braille under the pb directive.

Number pages with Roman Numerals: nr0 The nr directive lets you number pages using Roman numerals. To use Roman numerals when pages are numbered, use nr1. To turn off numbering in Roman numeral mode, use nr0.

Left Justify Text: lj yes ProWORDS assumes that you want your text left justified. If you use another justification directive, like center, you use the lm directive to reset left justification mode. Left justified text means that all your lines (except those indented at paragraphs) are even. Left Justified text has a ragged right edge.

Right Justified Text: rj no The rj directive is used to begin formatting your text with an even right edge and a ragged left edge. Use lj to return to text that is left justified.

Center Justified Text: cj no The cj directive is used to begin centering a line or lines of text. Use the lj directive to turn centering off and return to left justification.

Center Alignment: tc The tc directive tells ProWORDS how it should align centered text. If you use ct1, ProWORDS uses the values from the page size specification used in the last dt directive. (You must first use the dt directive to set the page values.) If you want to make center justified text align with the page and titles align with the text, use dt to define the page, then turn it off and use the ct directive to make centering use the page values from the dt directive. The default is to make center justified text align with the rest of the text on the page. It is recommended that you not use the tc directive until you become familiar with the system.

Full Justified Text: fj no The fj directive is used to make your text have an even left and right edge. ProWORDS accomplishes this by padding spaces between words. Return to left justification with the lj directive.

Skipping Lines: sk0 Use the sk directive to skip lines in the printout. Note that you can also skip lines with carriage returns. Multiple carriage returns, however, get absorbed, by default, when printing in braille. So, if you definitely want to skip lines on a printout no matter the method of printing, use the sk directive.

Command Character: cc . The command character precedes every directive. It is usually the PERIOD character but you may change it. The new command character should immediately follow the cc. So, to change the command character to an EXCLAMATION POINT, your directive would be cc!. After executing this directive, all the following directives would be initiated with an EXCLAMATION POINT character. We have, for instance, changed the command character to an EXCLAMATION POINT to print the directives without actually executing them for these pages. To change the command character back to PERIOD, type "!cc."

New Page: np The np directive forces any text following the np directive to begin printing on a new page. Do not embed the formfeed character into your text. ProWORDS controls the printing process without them. Use np to force a new page.

Odd Page: op The op directive acts very much like the np directive except that instead of ejecting to the next page, the op directive ejects to the next odd page. This is used when you want to insure that a new chapter or section starts on the right side of a pair of pages.

Conditional Page: cp 0 The cp directive allows you to conditionally force a new page. The argument to the cp directive refers to the number of printing lines left on any page. If, for instance, you used cp4, the directive would force a new page only if, at that point, there were three or less printing lines left on the current page.

Pause: ps no message The ps directive lets you pause the printing at any point. You may include, on the ps line, an optional message that will be displayed on the screen. A typical example might be, "PLEASE INSERT THE COMPANY LETTERHEAD." If you have a multiple-file document that extends beyond one or two disks (or more drives than your system supports), you might wish to use a message like, "PLEASE INSERT THE NEXT DISK."

Continue: cn no The cn directive aids in the editing process. It is used primarily in conjunction with the calculate print position (APPLE-K) command with multi-file documents. It works by grabbing the values from the previous file (which ends with a ch directive) to inform the calculator that this is not the first file.

If you wish to calculate the printing position of the second or subsequent file in a multi-file document, follow these steps:

1. Load the first file of the document.
2. Use the pr9 directive at the beginning of the first file. The pr9 directive tells ProWORDS to "print" the file to nowhere, thus allowing it to quickly calculate the actual position without wasting time printing to paper or disk.
3. Print the file with the APPLE-P command.
4. Notice the file names as they are loaded from the disk.
5. When you reach the desired portion of the document, abort the printing by pressing the RETURN key.
6. Make sure there is a cn directive at the beginning of this file. (It does not hurt to have a cn directive at the beginning of every file following the first.)
7. Now, when you use the APPLE-K command, the calculator knows to begin counting lines and pages with the values left from the previous file.

Chain Files: ch <pathname> The ch directive is probably the most powerful directive you will use. This directive, which must always appear at the bottom of a file, chains another file to the printing process. The syntax of the ch directive is this:

```
.ch file.to.chain
```

WARNING: Never print your file with this directive at the end of it unless you have saved your text; the ch directive loads the specified file into memory replacing whatever text was there previously.

If you ever get the "PATH NOT FOUND" error message while ProWORDS is trying to load another file in the printing process, just manually load the correct file, then type cn at the top of it. When you press APPLE-P, the document will resume with all the former formatting values retained.

You don't necessarily have to specify the complete path name with the ch directive. Instead, you might wish to specify only the name of the file, leaving off references to volumes and directories. This provides the advantage that a group of related files may be easily moved from disk to disk or directory to directory. Make sure that the prefix is set to the directory that contains all the files when you print. If you use the selector to load the first file, the prefix is set; so you may proceed.

If you use ProBRAILLE to translate a series of files chained together with the ch directive, the translator asks you to tell it

where to put the translated files. Just type the name of another folder where you want them stored.

Single Sheets: ss 0 The ss directive is used to specify that you are using a printer that requires single sheet feeding. If your printer requires that you feed in the paper a page at a time, use ssl; otherwise, use the default of ss0.

Set Tabs: st 0 The st directive permits you to set up to 20 tabs for use with the inclusive tb directive. Separate each number specified with the st directive with a comma. The numbers to use with the st directive are measured from the current left margin. To reset the tab stops, use the st directive again with the new values.

Contents Item: ci 0 The ci directive is used to mark a title in your text for inclusion as a table of contents item. It works by using the next line of text from where the directive appears up to either the first carriage return or the first occurrence of two spaces on that line and copies this information, along with the page number where that item appears, into a temporary file. The name of the temporary file used by the table of contents generator is Q. Before you begin generating a new table of contents, therefore, you should either delete or rename any file on the disk named Q.

The ci directive uses one parameter, a number from 0 to 255. The number is used to control the indention level as the contents file is generated. An inclusive tab directive is generated for each level specified. If, for instance, you use ci2, your table of contents file will contain two tab directives before the text of that title.

Each entry in the table of contents includes an indenting level, the text of the title, another directive that tabs to the right margin, and the appropriate page number where that title appears in the text. If you don't want any tab level on an item in the table of contents, you may either omit the parameter or use 0 as the tab level.

Generating the Table of Contents. The actual table of contents file is generated when you print your document. During testing of this feature, MicroTalk staff quickly realized that you don't necessarily want to generate the contents for a manuscript every time it is printed. A directive, therefore, is placed in the beginning of your text that activates the table of contents generator. It is recommended, in fact, that you first print your work, then use the pr9 directive along with the dcl (do table of contents) directive. The pr directive is used to specify the slot where your printer is connected. Using 9 as the parameter tells ProWORDS to print nowhere. Printing nowhere, in this case, may be just what you need, because it permits ProWORDS to process the text

according to the directives without wasting time with slow printers.

Once the table of contents is created, you should print it. Before printing, however, look it over; you may wish to edit or otherwise enhance the file. A macro has been supplied with the OPTION-M command that performs some usually needed functions on the file generated by the table of contents creating process.

Emulation Mode: em 0 The em directive permits printing files that were created with a version of ProWORDS earlier than version 1.4. The em1 directive tells ProWORDS that the underline character should be interpreted as a token to toggle between underlining on and underlining off. Notice that you should use the bu (begin underlining) inclusive directive in any new documents you create. This characteristic was changed in order to maintain compatibility with translating files into grade II braille.

Force Upper Case: uc 0 The uc directive is used to force every character in your printout into upper case. There are few times when you will want to do this. One time this directive comes in useful, however, is when using the Ohtsuki printer in the mode that both prints and brailles. The Ohtsuki printer requires, if you want both print and braille, that all characters be sent in upper case. The uc1 directive provides this function. Turn off the uc directive with uc0.

Orphans and Widows: or 1 The orphans and widows directive prevents the printout from containing breaks at page boundaries when the bottom line of the page contains only the first line of a new paragraph or the top line of a page contains only the last line of a paragraph. You may turn off the orphans and widows directive with or0.

Inclusive Directives

The following chart shows the inclusive directives. Many of the inclusive directives are used to control special formatting specifications that must occur within a line of text. When ProWORDS processes an inclusive directive that affects the way your printer treats the characters, it sends a code to your printer to select that mode. Obviously, your printer must support the functions that are requested. If, for example, your printer does not print italics, the bi (begin italics) directive will not make your work print in italics. Before you can use the control codes, you must set them to your specific printer. See your printer's manual for details.

Setting Printer Control Codes

The codes sent to your printer to control special formatting commands are set using the printer codes menu accessed with APPLE-ESCAPE.

When you press APPLE-ESCAPE, ProWORDS shows you several options on control codes. The menu looks like this:

Printer Codes Menu

Underline	none defined
Underline off	none defined
Boldface	none defined
Boldface off	none defined
Superscript	none defined
Superscript off	none defined
Subscript	none defined
Subscript off	none defined
Italics	none defined
Italics off	none defined
Serial init	IX E
Printer init	none defined

Use the ARROWS to select then press RETURN

Use ESC to return to editing

The right-hand column of the display shows the current control codes defined for each function. If none is defined, the words "none defined" appear in that column. If a printer control is defined, it is shown. CTRL characters will be displayed in inverse video.

Use the UP and DOWN ARROWS to move to the item to set. Press RETURN and type the exact code to send to your printer when the printing format is requested. Use the SHIFT-6 to tell ProWORDS that you are done typing the control codes. If you make a mistake typing a control code, press SHIFT-6 and try it again.

In most cases, you don't need to set the codes for bold face and underlining. These are handled in a generic fashion that works for nearly every printer. If you find that your printer isn't underlining correctly, change the underlining control command.

Remember that each of the inclusive directives begins with the CTRL-BACKSLASH character and each ends with the CTRL-RIGHT BRACKET character.

You might wonder why it is even worth bothering with inclusive directives when you can type your printer's control codes directly

into your text. There are at least two reasons. First, typing CTRL characters takes up space on the line. If you had very many formatting commands on a single line, the CTRL characters could cause the printing characters to end too early. Secondly, using the inclusive directive system permits your file to be printed on any printer. If you directly embedded the printer control codes into your text, you would have to edit each of them to use a different brand of printer. Using the directives permits you to simply change the definitions once to print your work on other printers.

Table of Inclusive Directives

Begin Underlining: bu The bu directive is used to begin underlining. If the text is translated into grade II braille, the underlining is appropriately translated. If printing in ink, the text is underlined by printing the character to underline, a backspace, and then the underline character. This works for all printers except those that do not support backspacing. If you find that underlining does not work with your printer, use the printer codes menu to specify the codes which select underlining for your printer.

Finish Underlining: fu The fu directive is used to indicate that you are finished underlining text.

Begin Boldface: bb The bb directive begins printing your text in boldface. This is done, in ink, by printing the character four times. If you see that instead of printing the character four times on top of itself, the character actually appears four times in your text, use the printer codes menu to set the code that your printer uses for boldface printing.

Finish Boldface: fb The fb directive finishes boldface printing.

Begin Subscript: b- The b- directive begins printing text in subscript.

Finish Subscript: f- The f- directive finishes subscript printing.

Begin Superscript: b+ The b+ directive begins printing superscript characters.

Finish Superscript: f+ The f+ directive finishes superscript printing.

Begin Italics: bi The bi directive begins printing in italics.

Finish Italics: fi The fi directive finishes printing in italics.

Tab to Next Tabstop: tb The tb directive forces the next text to print at the next tabstop. Set tabstops with the st directive

discussed in the "Directives" section of this manual. One thing to remember about tabstops is that their position is set relative to the left margin.

Tab to Right Margin with Specified Character: `tr[character]` The `tr` directive fills the line with the specified character. It works by looking for the next space or carriage return in your text. You may, therefore, use the `tr` directive to print items like a table of contents. Use the `tr` directive like this:

```
tr.35
```

This directive makes the number 35 appear on the page's right margin. All the space between the text immediately preceding the `tr` and the 35 is filled with periods.

Printing in Grade II Braille

There are several issues to consider when printing in braille. These range from differences in the codes used to represent the characters to the specifics of formatting.

One of the issues worth considering when you're ready to print in braille is the decision on which code to use. There are several including grade I, grade II, Nemeth, and Music braille. This discussion focuses on printing your work in grade II literary braille.

No discussion of braille is complete without a description of the basic braille unit, the cell. Experienced braille readers may wish to skip this description. The typical braille "cell" contains space for six possible dots. The dots are arranged in two columns of three dots each. They are numbered from top to bottom as 1, 2, and 3 and, in the second column, as 4, 5, and 6. Characters are represented by using different patterns of dots within a cell. In recent years, many computer users have lobbied for the use of eight-dot cells. They maintain that the eight-dot configuration provides more flexibility and enables the single-cell representation of capital and small letters as well as control characters. Others argue, however, that as long as the basic tool for writing and transcribing braille, the Perkins brailier, contains six dots, the chances of adopting the eight-dot cell standard are small.

The first step in braille printing is, of course, translating your file. Translating your work is necessary because the grade II braille code employs a different method of representing text than the more familiar "ASCII" representation used in your computer. ASCII stands for the American Standard Code for Information Interchange and is used by most microcomputers as a means of representing the characters we read. Grade II braille, on the other hand, is a code used for an entirely different purpose. One

of the chief objectives of the grade II braille code is to depict the text as accurately as possible in the least amount of space. To this end, grade II braille combines several often-used combinations of letters to form "contractions." The letter "T", for instance, represents the word "that" in grade II braille. It is the translator's job, whether it be human or a computer program, to replace the "that"s in your ASCII file with "T"s. Unfortunately, the translator cannot replace every "T" with "that". It does so only where the contraction is not ambiguous. Note that translating the "that" in "Thatcher" renders the translation inaccurate because the original word's meaning is unclear.

It is the translator's job to determine if the translation of a group of characters is accurate. To help with this task, the translator applies over 200 rules and over 400 exceptions to those rules in the process of converting your ASCII files into braille.

In addition to converting groups of characters into the equivalent ASCII code to represent a braille contraction, the translator must compensate for two shortcomings in the way braille embossers represent text. When it became clear that electronic braille embossing devices were practical, a group of authorities devised a code that attempted a one-to-one correspondence between ASCII and braille characters. For the purposes of this explanation, this code will be called ASCII computer braille. Do not confuse this term with the BANA computer braille code which is yet another, more accurate system of representing ASCII characters in braille. The ASCII computer braille code is the system employed by all braille embossing devices used with microcomputers.

ASCII computer braille can be most closely compared to grade I braille. There are, however, three major differences between ASCII computer braille and grade I braille. The first difference involves the fact that it requires two cells to represent several grade I characters. Think of capital letters and numbers. Grade I's capital letter is distinguished from its lower case counterpart by preceding the letter's symbol with a symbol to indicate capitalization. Similarly, numbers are represented with two cells, one for the number indicator, and one for the number. ASCII computer braille code solves the problem of distinguishing upper and lower case letters by ignoring it. There is, in short, no difference between upper and lower case letters in ASCII computer braille. Numbers are handled more elegantly. The system employed by the ASCII computer braille code to represent numbers resembles the Nemeth braille code which is used in the representation of mathematics. This system represents each number as its grade I counterpart's dropped cell and without the preceding number sign character. (Dropped cell refers to the condition where a character which uses only the upper two-thirds of the cell is represented, instead, with the lower two-thirds of the cell.) The number one is represented with dot two. (Its grade 1 counterpart is represented with a cell containing the number sign followed with a cell

containing dot 1.) The number two in ASCII computer braille uses dots two and three.

The second major difference between ASCII computer braille and grade I braille involves the representation of punctuation. The punctuation characters in grade I braille consist of characters formed with the lower portion of the cell. The comma character, for instance, is represented in grade I braille with a cell containing dot two. The period character is represented with dots 2,5,6. ASCII computer braille already uses these symbols to represent the numbers. Note that in ASCII computer braille the cell containing dots 2,5,6 represents the "4" and the cell containing the dot two represents the number "1". Since there are no contractions in grade I and ASCII computer braille, the problem of representing punctuation is solved by using the characters employed by grade II braille to represent contractions. A grade II's "TH" symbol is the ASCII computer braille code's question mark character. While the resemblance between the question mark character and the "th" sign seems random, most symbols for punctuation marks were carefully selected and often pictorially represent the respective character. The left parenthesis character, for example, is represented with the grade II symbol for the contraction "of" and the right parenthesis character is represented with the grade II symbol for the "with" contraction. The braille reader immediately recognizes that these symbols enclose the parenthetical text much as their ink print counterparts. You can examine the accompanying chart which indicates each ASCII character's ASCII number, the character, the way the Echo synthesizer pronounces the character, the dots used in a braille cell to represent the character, and the grade II meaning of that character. The material in the chart is not essential to successful grade II translations, it merely provides the more curious reader additional information.

The third difference between ASCII computer braille and grade I braille becomes evident when you realize that there are more ASCII characters than there are grade I characters. Note, for instance, that there are no grade I representations for the tilde and grave accent characters found in the ASCII character set and on your computer's keyboard. Interestingly, even with the omission of 26 of the alphabetic characters, there are more letters in the ASCII character set than there are remaining dot patterns to represent them. These remaining letters are handled rather unceremoniously. ASCII computer braille makes absolutely no distinction between the left brace, left bracket and right brace, right bracket character pairs. While such distinctions seem trivial to the normal grade II user, remember that ASCII computer braille's purpose is to represent ASCII characters. In this instance, it fails. Fortunately, the more extensive BANA computer braille code handles these obscurities. Although a translator is required to obtain BANA computer braille, you can feel fairly secure that in critical

cases where such distinctions are required, a translator is usually available.

Aside from the differences between ASCII computer braille and grade I braille discussed in the previous paragraphs, the two systems are very similar. The slight drawbacks of ambiguity when reading alphabetic characters is a minimal price to pay for the flexibility of obtaining a printout at any time without the requirement of translation software.

You will, of course, want to translate most text you write. Fortunately, translating software takes care of this tedious task without involving you. As the transcribed copy is a braille reproduction of an original text, the braille reproduction of your file does not affect the original document in any way. The translating software, in other words, creates a new file which contains the braille version of your original file.

After grasping the intricacies of the translation process, the alert user will see that the formatting of braille documents varies greatly from that of ink printing. Obviously, the size of the braille page is different from the ink page. Think too, however, about the differing requirements of margins, blank lines, page numbering, and paragraph indention. While the standard ink page holds a possible 66 lines of text, the braille page holds 25. Ink documents contain top and bottom margins of six lines or about one inch. Braille, on the other hand, uses top, left, and bottom margins of zero! There are, furthermore, only 40 characters per braille line. While such adjustments in the standard values you usually use for ink are automatically altered by your software when printing in braille, it is interesting to note the considerations that must occur. Note, too, the adjustments required in formatting to accommodate printing braille page numbers. The braille page number, in an effort to conserve space, is usually printed at the end of the first line of text. This is never done in ink.

Along with margin, page length, and numbering considerations, the software formatting your document needs to adjust the amount of space that is indented at paragraphs. Braille paragraphs are usually indented with two spaces. Ink documents almost always use five spaces. Depending on the particular format in use, ink pages sometimes leave blank lines between paragraphs. Braille pages almost never employ this style. Again, the software takes care of this step automatically, the differences, however, are interesting to note.

Specifying Braille

You might wonder how ProWORDS knows if you intend to print in ink or braille. The answer is that you tell it to do so by using a printing directive. Recall from the discussion on directives that these codes are used to control the way your text is printed.

While you actually type them into your text, they do not get printed.

The directive to inform ProWORDS that you are printing a braille document is `.pb` (for print braille). When you use the `.pb` directive, ProWORDS assumes that your top, bottom, and left margins are all zero. It assumes that you don't want to print any characters beyond the 40th column, and it assumes that your paper is only 25 lines long. Further, ProWORDS begins numbering your pages on the right end of the first line on the page. Another assumption ProWORDS makes when you use the `.pb` directive is that you want to suppress multiple carriage returns. The program, in other words, ends a line of braille when a carriage return appears in the text, but it ignores all subsequent carriage returns except for those appearing before a directive, thus avoiding large sections of blank text. By strategically placing your directives, you can choose to either print a blank line or not with the carriage suppression feature at work. If, for instance, you want to separate parts of a document with a blank line, press RETURN twice at the end of a paragraph or section of text. If the next line contains a normal directive (the line begins with the directive command character, the period), there will be a blank line printed in the text. If, however, you need to use a directive and want the print to skip a line but not the braille, type the directive immediately after the carriage return ending the paragraph. Then, press RETURN twice.

As you can see from the differences that the `.pb` directive makes in the way your text is printed, it is an important directive to know. Fortunately, that's all you need. You don't even have to type the directive. When you translate your file using the APPLE-G command, ProBRAILLE automatically adds the `.pb` to the beginning of your document.

Changing Standard Values

If there are any assumptions that are not exactly what you want from the use of the `.pb` directive, you have two courses of action. You may either change them by typing more directives or you may inform ProBRAILLE to automatically add as many directives as necessary to fit your situation. Some braille embossers may use a 27-line page; or, you might use slot two as the printer port. By adding the `.pl27` and `.pr2` directives to your table of automatically typed directives, these codes are placed into every document that is translated with the APPLE-G command.

To alter the translation table to insert other directives, load the file TBL2 into ProWORDS. Save it on another disk before you do another thing. Try saving it as OLD.TBL2 to keep track of the original file.

Notice that the beginning of the file starts with a period and a carriage return. The next entries in the table are ".pb" and a carriage return. After this carriage return, add as many directives as you wish to fit your exact needs.

Consult the "Printing" section of this manual for complete details on directives.

The following list of braille oriented directives supplies the details of each directive's use and parameters. Remember that all normal directives (those which are not of the inclusive type) either appear as the very first item in the file or each appear on a separate line.

Braille Oriented Directives

Function Code Default Description

Print in Braille: pb no default The .pb directive tells ProWORDS to format your text for braille printing. This means that ProWORDS changes its assumptions about the standard values of formatting items. Instead of a left margin of 12, for instance, the .pb directive makes ProWORDS assume that the left margin should be 0. In addition, ProWORDS assumes that the top margin is 0, the bottom margin is 0, the page length is 25, that you want to suppress multiple carriage returns except where directives appear, and that you want each page numbered on the first line. You may, of course, change any of these assumptions by using the directive that corresponds to the item to change.

Besides making different assumptions about your print format, ProWORDS is effected by the .pb in another way. If the .pb directive is present, ProWORDS changes the way page numbers are printed. Instead of printing out the normal page number, ProWORDS translates the number, on the fly, into grade II braille and then prints the number sign character and the translated page number. What this means is that when you use the .pb directive, ProWORDS prints the text in memory as if it were braille.

Along with changing basic assumptions and forcing an on the fly translation of page numbers, the .pb directive has one more effect worth noting. When used in conjunction with the .ci (contents item) directive to generate a table of contents, the .pb directive also translates the page numbers that appear in that table. Since the table will presumably be generated from a document that is already translated, it is important to have these page references created in this manner.

Braille Numbering: nb 1,0 See the section "Braille Page Numbering" in the "Directives" section of this manual.

Suppress Multiple Carriage Returns: sc 1 The .sc directive is used to suppress multiple carriage returns, thus preventing large blank spaces in braille. The default condition, when the .pb directive is used, is to suppress multiple carriage returns. If you would not like multiple carriage returns suppressed, use the directive .sc0 to turn this function off.

If you wish to create your text so that it prints a blank line between paragraphs when printed in ink and not skip lines when printed in braille, just press RETURN twice at the end of a paragraph. There are times, however, when you will want to leave a blank line between sections of text even in braille. If ProWORDS sees that the next character after a multiple occurrence of a carriage return is a directive, it skips the line even in braille. This proves useful for separating sections of text. If you want to use a directive between paragraphs and don't want a blank line, type the directive or directives that are to effect the next paragraph immediately after the first carriage return that ends the previous paragraph. After typing the directives, press RETURN twice. This leaves your text in shape so that in ink, a line is skipped while in braille, it is not.

Setting the Token Value: tk 2 or 5 The tk directive is used to set the value of the token character. The token character is used to specify one of two different numbers depending on the mode of printing-braille verses ink. The token character is the numbersign, typed by shifting the 3. The value of the token character is either 2 for braille or 5 for ink. You will probably never need to change these values but you may do so if needed with the tk directive.

The tk directive takes two parameters. Each is a number which represents the value to assume and each is separated with a comma. The first number specifies the value for ink printing. The second number changes the value for braille printing.

Inclusive Braille Oriented Directives

* Note that the following directives are embedded into your text. Each is preceded with the CTRL-BACKSLASH character. The CTRL-BACKSLASH is pronounced "BOUND" because it is used as a boundary between your normal, printing text and the inclusive directive. Each contains two characters to indicate the specific command, and each may optionally have parameters. Each inclusive directive ends with the CTRL-RIGHT BRACKET character. ProWORDS pronounces CTRL-RIGHT BRACKET as "UNBOUND" because it marks the end of the directive.

The following table indicates the command characters and any optional parameters that are used with each inclusive braille oriented directive.

Function Code Description

Do Braille Only: db The db directive prints the text up to the next ed directive only if you are printing in braille. You can use this directive to include text or formatting commands that should be used only for braille printing.

Do Print Only: dp The dp directive prints the text up to the next ed directive only if printing in ink. This directive is similar to the db directive except that the material is printed only in ink mode.

End Do Section: ed The ed directive ends a conditional do section. The conditional do sections are initiated with either the db or dp directives. The ed directive ends either ink or braille conditional do sections.

Do Single Braille Cell: ds The ds directive prints a single braille cell. The only parameter this directive requires is the character to print. The character to print will not be translated, so be sure to include the exact ASCII braille cell to be embossed.

The ds directive is useful to highlight or clarify an ambiguous character by surrounding it with full cell characters. The ASCII representation of the full cell character is the equals sign. A directive that prints a single equal sign in braille only looks like this:

ds=

You will notice that ProBRAILLE translates some letters which appear by themselves as the letter sign then the letter. Others, those which are not ambiguous in particular, are translated as the letter without the letter sign. Such letters include "A", "I", and "O". You may use the ds directive with the semicolon as a parameter to force ProWORDS to print the letter sign with the letter.

Grade or Level of Translation: gr The gr directive selects the grade or level of translation. It is necessary to use this directive only if you wish to change the translation level. By default, ProBRAILLE translates your file, when you access it via ProWORDS, into grade^_II braille. You may, however, change it to either no translation, grade 1 translation, or BANA Computer Braille. You select the grade of translation by including a number with the directive. The numbers to include are 0 for no translation, 1 for grade 1, 2 for grade 2, and 3 for BANA Computer Braille.

The Emphasis Symbol: bu The bu directive is the same as the ink begin underline directive. When translated, ProBRAILLE marks the beginning of the section of text to be emphasized or underlined

with the proper braille characters and marks the ending of the section with the appropriate braille character for ending the emphasis. If you print the translated document, ProWORDS obviously ignores the bu directive because the underlining has already been taken care of by the translator.

Transcriber's Option: to The transcriber's option directive takes one of two parameters; either 1 or 2. The to1 directive places the transcriber's option 1 symbol into your text. Similarly, the to2 directive puts the second transcriber's option symbol into your text.

English Braille Mode: eb The eb directive instructs ProBRAILLE to begin translating in English Braille. English Braille does not use the capitalization symbol.

PART III: ProWORDS Reference

Cursor Movement

LEFT ARROW moves the cursor one character left.
RIGHT ARROW moves the cursor one character right.
APPLE-LEFT ARROW moves the cursor one word left.
APPLE-RIGHT ARROW moves the cursor one word right.
OPTION-LEFT ARROW moves the cursor back to the previous carriage return.
OPTION-RIGHT ARROW moves the cursor to the next carriage return.
UP ARROW moves the cursor to the previous sentence.
DOWN ARROW moves the cursor to the next sentence.
APPLE-UP ARROW moves the cursor one screen up.
APPLE-DOWN ARROW moves the cursor one screen down.
OPTION-UP ARROW moves the cursor to the beginning of text.
OPTION-DOWN ARROW moves the cursor to the end of text.
APPLE 1-9 moves the cursor to the previously set mark.
OPTION 1-9 sets marks to which you can later return with APPLE 1-9.
CTRL-B moves the cursor to the beginning of a sentence or back to the first carriage return.
CTRL-F (followed by text to find) moves the cursor either forward or back to text you want to find.
CTRL-G makes the cursor go forward by words until you press another key.
CTRL-R makes the cursor move by sentences, reading on the way, until you press another key.
CTRL-S (followed by search and/or replacement text) moves the cursor to text for which you search and/or replace.

Deleting Text

DELETE deletes the character left of the cursor.
APPLE-SPACE recovers and inserts the last deleted character.
APPLE-DELETE deletes the word left of the cursor.
OPTION-SPACE recovers and inserts the last deleted word.
APPLE-X deletes back to and including the previous carriage return.
OPTION-DELETE deletes from the cursor back to mark 1.
APPLE-OPTION-DELETE deletes from your cursor to the end of the text.
CTRL-D (followed with another character) deletes forward to the first instance of the specified character in the current paragraph.
CTRL-N stands for "new" and completely removes text from the workspace.

Move and Copy

CTRL-C copies the block of text between mark 1 and mark 2 and inserts it at the cursor; then, it asks if you want to delete the original block.

Information and Preferences

CTRL-E sets Echo TEXTALKER or ProSCAT commands.
APPLE-F announces free workspace.
CTRL-L enters TEXTALKER's line review mode.
CTRL-O lets you type another character of any kind.
CTRL-P toggles the clicking (popping) of keystrokes.
CTRL-Q makes the arrow keys change from moving by sentences to going to the next RETURN.
CTRL-T announces the current time.
CTRL-V toggles voicing of keystrokes.
CTRL-6 toggles between upper case detect and normal speech.
CTRL-UNDERLINE inserts a sticky space.
APPLE-CTRL-A shows the ASCII code of the character left of the cursor.
APPLE-GRAVE ACCENT inserts the current date at the cursor.
APPLE-TILDE inserts the current time.
APPLE-LEFT BRACKET saves your preferences.

Disk Functions

APPLE-A appends workspace to a file on disk.
APPLE-B saves a block of text between mark 1 and the cursor to a file on disk; if the file exists, the workspace is added to the end of the file.
APPLE-C copies a file.
APPLE-D deletes a file.
APPLE-E executes another program.
APPLE-G translates a file into grade II braille.
APPLE-I inserts a file at your cursor.
APPLE-L loads a file from disk into the workspace.
APPLE-S saves the workspace to disk.
APPLE-M makes a new folder.
APPLE-N changes the name of a disk, folder, or file.
APPLE-O lets you load text that didn't completely fit into your workspace.
APPLE-Q quits the program.
APPLE-R receives text from a device connected to one of your serial ports
APPLE-CTRL-R resets port to use for receiving.
APPLE-T transmits a file to a device connected to one of your serial ports.
APPLE-U unlocks and locks files.
APPLE-V lets you view the contents of another file.
APPLE-SLASH shows you a list of files on the disk last used.
APPLE-QUESTION MARK shows you a list of files in the open folder.
APPLE-COMMA lets you open another folder.
APPLE-PERIOD lets you select another disk to use.

Printing

APPLE-P prints your work.

APPLE-ESCAPE lets you set interface card and printer parameters.
APPLE-K calculates and displays your work on the screen as it will print.
APPLE-J just tells you the page, line, and column number of your cursor if your work were printed with the selected formatting commands.

Directives

Note: the following normal directives always appear on a line by themselves and begin with the period character.

pr selects printer slot.
pp selects amount of indentation on paragraphs.
sp sets spacing between lines.
tm sets top margin.
bm sets bottom margin.
lm sets left margin.
rm sets right margin.
pl sets page length.
sk skips lines.
cc changes the command character.
cj center justifies.
tc makes centering align with the page rather than the margins.
lj left justifies.
rj right justifies.
fj fill justifies.
ti sets one of seven different titles.
dt makes titles align with page defaults rather than margins.
cp conditionally breaks a page.
np moves to a new page.
op moves to the next odd page.
ot sets odd titles.
pg sets page number.
nr numbers pages with Roman numerals.
PB prints in braille mode.
nb numbers pages in braille format.
st sets up to 20 tab stops. Each number is separated with a comma.
tk sets the token value.
cw sets the carriage width.
ch chains another file.
dc generates the table of contents (provided the ci directive has been used in the text).
ci designates that the following line is considered a contents item. This is used in conjunction with the dcl directive.
uc forces all text in your file to upper case.
em enters emulation mode.

Inclusive Directives

The following inclusive directives all begin with CTRL-BACKSLASH and end with CTRL-RIGHT BRACKET.

tb tabs to next tab stop.
bu begins underlining.
fu finishes underlining.
bb begins boldface mode.
fb finishes boldface mode.
b+ begins superscript.
f+ finishes superscript.
b- begins subscript.
f- finishes subscript.
bi begins italics.
fi finishes italics.
tr tabs to right margin with specified character.
gr sets level of braille translation.
db does only printing in braille mode.
dp does printing of following text only in ink mode.
ed ends conditional printing.
ds does printing of a single character in braille mode only.

PART IV: ProTERM

Professional Talking Terminal

INTRODUCING ProTERM

Congratulations. You chose the most powerful talking terminal available for the Apple II series of computers.

ProTERM is an intelligent talking data communications program. It transforms your Apple //e, Apple //c, or Apple IIgs into a talking terminal for nearly any remote computer system or special device. Since ProTERM is an intelligent terminal, you can transmit and receive files with the remote system. In short, you can use the convenience and power of your favorite word processor to create and edit text, then when you are ready, send it all at once to another computer. You can also gather large amounts of information and save it for later reading when you aren't being charged to do so.

Setting Up

As described in the introduction to this manual, ProTERM requires that you use either the built-in ports on the //c and IIgs or a Super Serial Card on the //e. (The serial port provides an interface between your computer and the modem or other device.) ProTERM also requires that you use an external modem. It does not function with modems that plug directly into one of your computer's slots. With the price of external modems down to as low as \$150 for a 2400 baud model and \$70 for 1200 baud, this should prove no inconvenience. It is the author's experience that external modems provide not only more flexibility, but prove easier to use as well.

While it doesn't matter which slot you choose for installing your serial interface card, the standard place is slot 2. Apple //c users, however, will have to plug the modem into port 1 if they are using the Apple //c with a Cricket synthesizer. Port 1 is generally used for printers, but it works just fine with modems, too. (The Cricket synthesizer must stay connected to slot 2.)

Notes About the Super Serial Card If you are using an Apple //e or if you plug the Super Serial Card into a IIgs, the Super Serial Card must be configured properly for communications mode. There are two things to consider when configuring the card--the jumper block and the DIP switches. The jumper block is a chip with a triangle on it. Above and below the jumper block are the words "terminal" and "modem." The jumper block on the SSC should be positioned so that the triangle points to the word MODEM. If the triangle points to TERMINAL, carefully remove the chip and flip it so that the triangle points to modem. When the jumper block is properly positioned, turn your attention to the two sets of DIP switches. Each group is called a "bank" of switches. The first

bank is closest to the left edge of the card. The switches on the SSC should be set as follows:

Super Serial Card Switch Settings

ON OFF OFF ON ON ON ON (on the first bank)

OFF OFF OFF ON OFF ON OFF (on the second bank)

Note that //c and IIgs owners need not worry about the switch settings. ProTERM automatically configures the built-in serial port as required for the //c and IIgs.

Getting Started

Choosing Terminal from the main menu starts the terminal program. When ProTERM first starts, it looks for a serial interface in your computer. It assumes that the interface is connected to slot 2, the standard place for modem use.

Note to //c Users

Note that if you use an Apple //c with the Cricket synthesizer, the first thing you'll want to do is tell ProTERM to use the port in slot 1 (the printer port) for your modem work. The Cricket must be connected to slot 2. You can do this by pressing APPLE-P to access the parameter menu. Next, press C to select communications port then press 1 to select port 1. See more about parameters in the following paragraphs.

If your serial interface is connected to a slot other than 2, ProTERM shows you the parameter menu and requests that you show it which serial interface to use. If you have only one serial interface, ProTERM shows you where it is and asks that you confirm its use by pressing RETURN.

TERMINAL MODE

Terminal mode is where you communicate with other devices. ProTERM automatically puts you into terminal mode when you first start the program so that, as soon as you begin, you are ready to communicate. In terminal mode, your Apple lets you communicate with nearly any other device that supports serial data communication.

Going Online

When you start the program, ProTERM is ready to go. Most everything you type at your keyboard is sent out the serial port to the other device to which you are connected. Usually, the other device is a modem which in turn connects your computer via the phone lines to another computer, but you can also communicate with

a variety of devices that support serial communications. Optical character readers, other personal computers, and some of the new, small note-takers like the PocketBraille and Braille 'N Speak are examples.

Commanding the Program While most characters you type are sent directly out the serial port for the remote device, there are times when you want to tell your system, ProTERM in this case, something. You give ProTERM commands by holding down the APPLE key while pressing another letter. ProTERM also uses some of the CONTROL commands, accessed by holding down the CONTROL key while typing another letter. (There is also a way to tell ProTERM to send that CONTROL character to the serial port.)

ProTERM uses commands to let it know a variety of communications modes, preferences, and to initiate the transfer of large amounts of data. Pressing APPLE-P, for example, accesses ProTERM's parameters menu. Pressing APPLE-R lets you receive files from other computers. You can read more about those commands shortly.

Commanding your Modem Your modem uses commands to tell it a variety of preferences and to take a variety of actions. One of the most frequently used modem commands is the dial command. If your modem is Hayes compatible, and most of them are, the commands it accepts all take the form of an attention getting sequence followed by the actual command.

The attention getting sequence is simple. It consists of the letters A and T. If your modem is an old one, it may require that these letters be typed in upper case.

After the attention sequence, the modem wants another letter to tell it what command to perform. In the case of the dial command, that letter is, appropriately enough, D.

After you type the attention getting sequence and the command letter, many of the modem commands also take parameters to the command. The parameter in the case of the dial command is the number you want to dial.

You should consult your modem's manual for all the commands available. There are some handy ones that let you set up touch tone dialing, pausing for a second dial tone, and, on some modems, establishing an extended result code sequence that proves quite useful. "Appendix E" of this manual summarizes many of the more useful Hayes compatible commands.

Preparing to Establish the Connection Before calling another computer, take a moment to read the section of this manual called "Setting Parameters." This is important because you must make your computer match the parameters required by the system you call. ProTERM's defaults should work with many systems you call, but take

the time to read this section anyway. You will learn ways to adjust the speed at which data is transferred possibly saving you lots of money.

Establishing the Connection Once you have typed the dial command and the number, press RETURN to dial the phone. You should actually hear your modem pick up the phone and dial the number and the remote system answer and send its "carrier signal". [A carrier signal is a tone which the modem modulates to create one of two different sounds for data transmission.] At this point, by default, your modem's internal speaker shuts off and the modem gives up control of its "command state" and enters "online state". In online state, you can no longer issue modem commands. Instead, your modem sends everything you type directly to the other computer.

Working Online When the remote computer answers the phone, you should see a message like this:

CONNECT

This is the modem's way of informing you that it has indeed detected a carrier signal from the remote system and has entered online state. Remember, once the modem enters online state, you may no longer issue commands to the modem. Instead, everything you type is passed on to the remote system just as if your Apple were a terminal to that system. Your Apple with ProTERM, however, does much more than just a terminal. You, in the following sections, will learn how to access your Apple's disk drives so that you may capture and transmit data from files with the remote computer.

Most computers require that you press RETURN once or twice after you get the CONNECT message. That lets them adjust their baud rate to the baud rate at which you called.

Once you've pressed RETURN, the remote system usually asks you for an ID or to type something like "NEW" if you are a new user. If you connect with one of the pay services like CompuServe or GENIE, you'll have already had to make arrangements with the company to establish methods of payment. At that time, the company assigns you an ID and password. The ID is used as your name on the system. Others will use it to send you mail. Your password, on the other hand, is private. Never give your password to anyone.

If your computer displays the "CONNECT" message, you will notice that, with each character received, ProTERM clicks your Apple's speaker. This lets you tell exactly when data is actually being transmitted. The data transmission, especially at 1200 and 2400 baud, occurs far faster than the screen can scroll. The speech also continues long after the actual transmission is complete, especially at higher baud rates. ProTERM provides a means of instantly catching the speech up to the actual data. While you may

not want to use it all the time, it offers a way to maximize your connect time. Use CTRL-D to make the speech catch up when the clicking quits. You can read more about this command in the "Dumping the Speech Buffer" section of this manual.

Setting Parameters

Before you can successfully communicate with another system, both systems must agree on the format of the characters they will transmit and receive. The following paragraphs discuss the way you tell ProTERM what format you want to use.

Use APPLE-P to access the parameter menu. This menu displays the current settings and lets you adjust the settings to your taste and to the requirements of the other device. It looks like this:

Parameters

Communications Port	<u>1</u> 2 3 4 5 6 7
Baud Rate	<u>300</u> 1200 2400 4800 9600
Data Bits	7 <u>8</u>
Stop Bits	<u>1</u> 2
Parity	Even Odd <u>None</u>
Half Duplex	Yes <u>No</u>
Keystrokes Voiced	<u>Yes</u> No
Incoming Data Clicking	<u>Yes</u> No
Mode for HAM Radio	Yes <u>No</u>
Upload Prompt	<u>None</u> ! @ # % ^ & () - _ = + [] < >
Outgoing Line Length	0 10 20 30 40 50 60 <u>70</u> 80 90
Line Length Incoming	0 10 20 30 40 <u>50</u> 60 70 80 90
Terminal Line Length	0 10 20 30 40 50 60 <u>70</u> 80 90
Returns Delayed	<u>Yes</u> No
VT100 Emulation	Yes <u>No</u>
Add Linefeeds	Yes <u>No</u>

Use the UP ARROW and DOWN ARROW to move the highlighted bar from selection to selection. (You can also press the first letter of your choice.) Then, use the RIGHT ARROW and LEFT ARROW to adjust the selected item. (You can also press the first letter of the adjustment.) In other words, to set baud rate at 2400, you can press "B" to move the selection bar to the Baud Rate selection, then press "2" to set the baud rate to 2400.

Communications Port The Communications Port selection on the parameters menu shows you a possible 7 places in which serial ports may appear. You will only be permitted, however, to choose ports where you actually have serial cards. If you use the RIGHT ARROW and LEFT ARROW keys, only numbers where you actually have serial ports will be highlighted. If you try to select a port not actually installed in your computer by typing the number of the

port, the menu buzzes and leaves the highlighting on the last port selected.

Apple //c and Apple IIgs users will note that there are two built-in ports. They are, even though the //c doesn't really have slots, treated as if they were in slots 1 and 2. The IIgs does have slots. You can use both the built-in port and a serial interface connected via one of the slots. You choose between them by changing the setting in the control panel from "Built-in Port" (for the port) to "Your Card" (for a Super Serial Card connected to the slot.) You can, of course, also use DAVEX's CONP command to switch from internal to external. The command to set slot 2 to external is "conp -sx2". Setting slot 2 back to internal (for the built-in port) is "conp -si2". Remember, whenever you change slot designations, you must re-boot your machine before the setting takes effect.

Baud Rate The Baud Rate selection choice on the parameters menu lets you adjust the speed at which you send and receive. There are currently no inexpensive modems that use baud rates above 2400 but the higher settings come in useful for downloading data from optical scanners, local computers, and special-access devices. If you are using such a device and you have an Apple //e, the SSC used to communicate with that device must be configured differently than the settings used for communications mode. In short, the jumper block must have its triangle pointed towards the word TERMINAL. An inexpensive alternative to using two SSCs or, worse yet, pulling out one SSC and constantly changing its settings, is to obtain a modem eliminator (null modem). (Turning the jumper block on the SSC to TERMINAL essentially performs the same task as a null modem.) The null modem changes two of the pins in the cable. Specifically, it crosses pins two and three. While it isn't critical that you know about the exact purpose of pins two and three, it is useful to know that they are the two pins that are used to send and receive the data. (Other pins are used for grounding and flow control.)

Note that, as with all the parameters discussed in this section, both systems must agree on the baud rate they use. Many remote systems, however, automatically lock onto the baud rate you are using when you first call the remote system. If this is not the case or if you are communicating with a local system, you must determine the parameters that the system uses then adjust ProTERM's parameters to match them.

Data Bits The Data Bits choice in the parameters menu lets you adjust the number of bits that each character uses. The choices are 7 and 8. The usual setting is 8.

Bits are electrical states that combine (either 7 or 8 of them) to make a character. Your modem turns these electrical states into sounds for transmission over the phone lines.

Stop Bits The Stop Bits choice on the parameters menu lets you specify the number of bits that appear between characters. The choices are 1 and 2. Most systems always use 1.

Parity The Parity selection on the parameters menu lets you decide the method of checking for valid data. Your choices are even, odd, and none. Most systems use none.

Parity is a means of insuring valid data. It is a scheme that adds another bit to your character. That bit's setting depends on which type of parity checking you choose. If you select even, that bit will be set so that there are an even number of set bits in a character. If you choose odd, the bit is set to make the number of set bits odd. If you choose none, the bit is not used.

Half Duplex The Half Duplex selection on the parameters menu lets you specify that ProTERM should put the characters you type on the screen. Most computers you call will echo the characters you type back, so you'll want to choose "No" here. If, however, you communicate with another personal computer or a system that doesn't echo your keystrokes, set Half Duplex to "Yes". If what you type comes out twice on your screen, set Half Duplex to "No."

Keystrokes Voiced The Keystrokes Voiced selection on the parameters menu lets you specify if the characters you type should be spoken. If you like to hear what you type, set Keystrokes Voiced to "Yes."

Note that in order for keystrokes to be voiced, they must appear on your screen. If you communicate with another personal computer or a system that doesn't echo back your keystrokes, you must have Half Duplex to "Yes" before Keystrokes Voiced will voice your keys.

Incoming Data Clicking The Incoming Data Clicking selection on the parameters menu lets you decide if you want to make your computer click with each character received. This clicking proves a valuable tool to keep track of exactly what is happening with your connection.

Note that as text comes in, your computer's voice falls far behind what is actually coming in. The clicking keeps you clued in to exactly when the data stops coming. As soon as it stops, you are free to type your next command to the system. Note that ProTERM makes the remaining information be placed on the screen without speaking it. This permits you to get about the business of moving to better places if you are already familiar with the menus the system uses. Of course, if you are new to the system, you won't want to press the next command as soon as the data stops clicking. Instead, you'll want to read the information and act accordingly.

Mode for Ham Radio The Mode for Ham Radio selection in the parameters menu lets you specify that ProTERM should spell out ham

radio call letters. It works by looking for words with digits in them. If ProTERM sees one, the word is spelled.

Spelling words like Ham Radio call letters makes the speech program user able to understand those call letters without having to use the review mode.

ASCII Upload Prompt The ASCII Upload Prompt selection on the parameters menu lets you upload text files to computers that pause between each line of text. These computers usually display some prompt character to let you know when it is safe to begin typing the next line of text. Instead, you can set the ASCII Upload Prompt to that character and send a previously prepared file to the other computer letting ProTERM pause while the other computer takes the information and signals its readiness for more. When ProTERM sees the prompt character, it sends the next line of text.

If you don't want to use prompted uploads, set this adjustment to "None."

If the remote computer fails to send the prompt character, you can press CTRL-Q to send the next line of text anyway.

Outgoing Line Length The Outgoing Line Length selection on the parameters menu lets you send text files to remote computers that require a carriage return at the end of each line even if your file doesn't have carriage returns. Use the ARROW keys to set the number of characters to send before ProTERM begins looking for a space. When ProTERM finds the first space after the specified number, it automatically sends the other computer a carriage return. If you have ASCII Upload Prompting set, ProTERM will also wait for the prompt after this carriage return is sent just as if you typed it yourself. '

Incoming Line Length The Incoming Line Length selection in the parameters menu lets you strip extra, unwanted carriage returns from incoming text. Many computers put a carriage return at the end of every line of text. Most of the time, this is undesirable.

The Line Length Incoming selection works by counting characters. If a carriage return comes in after the selected number of characters passes, ProTERM replaces it with a space. If the carriage return comes in before the selected number, it is preserved, thus preserving material that is meant to appear line-by-line.

Select 0 to preserve the exact format of the incoming text.

Terminal Line Length The Terminal Line Length selection on the parameters menu lets you tell ProTERM to ring a bell when your cursor reaches a specified screen position. This proves useful

when typing letters while connected to systems that require returns at the end of each line. To turn this feature off, select 0.

Returns Delayed The Returns Delayed selection on the parameters menu lets you make ProTERM wait for about 1.5 seconds after sending a carriage return when using ASCII uploads. This proves useful for systems that require a little time to react to a new line of text.

VT100 Emulation The VT100 Emulation selection on the parameters menu lets you emulate a VT100 terminal. This is useful for many business and university mainframe networks.

When you use VT100 emulation, ProTERM becomes a full-screen editor.

Keypad Keys To use the application keypad mode, press the OPTION key while you press the appropriate key on your numeric keypad. If you don't have a numeric keypad, the digits 0-9 and the punctuation COMMA, PERIOD, DASH, and PLUS on the main keyboard will also send the appropriate application keypad key if you hold down the OPTION while pressing it.

PF Keys You can send the VT100 PF keys 1-4 by holding both OPTION and SHIFT while pressing 1-4.

Add Linefeeds The add linefeeds selection on the parameters menu refers to adding linefeeds to carriage returns when using ASCII file transfers. When you have add linefeeds enabled, ProTERM adds a linefeed to each carriage return it sends to the remote device.

Set Clicker Volume on GS

Pressing APPLE-LEFT lowers the volume of the clicker. APPLE-RIGHT raises the volume. These commands work only on the Apple IIgs. (Older Apple IIs do not support setting the volume from software, although you can set the volume on a //c with the volume knob.)

Changing the volume is handy for those times when you are downloading a long file and want to leave the room. Raise the volume before you leave. You can monitor the file transfer from another room. When you return to the computer, use APPLE-LEFT to lower the volume.

Characters to Omit

Press Apple-O to specify any characters that you don't want spoken. This option proves useful when using systems that decorate the screen with characters that may be annoying with speech.

If you wish to include control characters in the characters to omit, you should precede each control character entered with a press of the CTRL-O. Pressing the CTRL-O lets ProTERM know that you intend to insert a control character into the string of

possibilities. (If you did not precede the control character with CTRL-O, ProTERM would act upon that character as if it were an editing command.)

Once you have typed the characters to omit, ProTERM returns to terminal mode.

Cancel Omitted Characters Press CTRL-APPLE-O to cancel all characters you have specified for omission.

Scrolling

You may use the APPLE-UP ARROW command to scroll back through text in the capture buffer. Each press of the APPLE-UP ARROW scrolls back about 255 characters. This command may be desirable after using the dump buffer command to examine the last portion of the capture buffer. If you continue scrolling until you reach the buffer's beginning, ProTERM buzzes to alert you.

Stopping and Starting the Remote System

Most computers support a form of flow control called XON/XOFF. The XON-XOFF flow control consists of sending the remote a character to tell it to stop, then another character to tell it to resume sending text. The character to send the remote system to tell it to stop is CTRL-S. CTRL-Q gets it going again. Unfortunately, CTRL-S is not a convenient character to use with TEXTALKER. It is the command used to tell TEXTALKER to stop, too. It is for this reason that ProTERM offers an alternate keystroke that sends the CTRL-S. That keystroke is CTRL-APPLE-A.

Quit

Press APPLE-Q to quit ProTERM. When you press APPLE-Q, ProTERM asks that you confirm your request by pressing Y for "Yes". It then automatically saves the capture buffer if you have turned it on then the program ends. Be sure to consult "Appendix C" for more information about the quit command and the other DOS commands ProTERM supports.

SPECIAL SPEECH FEATURES

ProTERM was designed with speech access in mind. There are, therefore, several features not found on other telecommunications programs. The incoming data clicker is one. This section of the manual describes others.

Silence Speech

ProTERM utilizes a "double-buffering" technique to allow you to capture data at the fastest baud rate yet maintain speech output. This means, however, that the speech can lag way behind the actual

data. You may silence the speech, allowing the speech buffer to catch up to the data buffer, with the CTRL-X command. Note that this is the normal for both TEXTALKER and ProSCAT to silence speech, so it shouldn't be hard to get used to. CTRL-X silences the speech until the next keystroke. In ProTERM, if you have silenced the speech, you may restore it without passing any character on to the remote system by pressing the SPACE BAR to re-enable speech. Any other character you press will also re-enable speech with one important difference. The difference is that any other character used to enable speech is also passed on to the remote system.

A Note About Keyboard Buffering on the IIgs Apple IIgs users should be careful when using the keyboard buffering feature. After silencing the speech with CTRL-X, you must use the IIgs's flush buffer command (CTRL-APPLE-DELETE) to re-enable speech. This is not a problem specific with ProTERM; it involves any program using TEXTALKER speech while the IIgs keyboard buffer is enabled.

Dumping the Speech Buffer

You may make the speech instantly catch up to the real data with CTRL-D. This command, in conjunction with the incoming data clicker, lets you cut your on line time to a minimum by using it when the clicking stops to put you where the action is.

If you want to send the remote system a CTRL-D, press CTRL-O then press CTRL-D.

The Arrow Keys

ProTERM sends the backspace character to the remote system when you type the LEFT ARROW key. In addition, you may use the RIGHT ARROW key to grab a character directly from the screen and send it to the remote system. This means that you can use your speech program's feature to exit review at a specific point to copy information from the screen. Get complete details on this procedure in either the TEXTALKER User's Manual or in the ProSCAT User's Manual.

You may already know that the CTRL-U is exactly the same as the RIGHT ARROW key. This means that, if you wanted to send the remote system a CTRL-U, you would, instead, send the character under the cursor. To send the CTRL-U, hold down the APPLE key while you press the CTRL-U.

Receiving Files

Press APPLE-R to prepare ProTERM to receive a file from the remote system. When you press APPLE-R, ProTERM offers you a choice of the kind of file transfer you would like to use. The kind you choose depends on several factors which are outlined in the discussion of each type.

The file transfer protocol menu looks like this:

Receive

ASCII

BINARY II

ProDOS

XMODEM

Use ARROWS to move
RETURN to select

As the menu's instructions indicate, use the arrow keys to move to the desired transfer protocol, then press RETURN to select it. (You can also press the first letter of the desired selection.) When you do, ProTERM asks you to enter the name of a file that will contain the transferred text or program. If you select the ASCII protocol, and the file currently exists, ProTERM announces that it is adding to the end of the material already stored in that file. If you want to ignore the information previously stored in that file, use APPLE-D to delete the file before using APPLE-R to receive another file with that name. If you choose one of the other protocols, and the desired file already exists, ProTERM requests permission to delete the contents of that file before beginning the transfer.

The protocol to use depends on what kind of material you want to receive. If you intend to capture all text that comes to the screen, use the ASCII protocol. If you intend to transfer programs, use one of the others.

Receiving ASCII Files

When you select the ASCII protocol, ProTERM acts exactly as it always does except that everything you type and everything that comes in through the serial port is recorded on a file on your disk. When ProTERM's internal storage capabilities are exceeded with incoming information, it sends a signal to tell the other computer to stop sending while ProTERM saves the information to the specified file. When the information is safely recorded, ProTERM clears its internal workspace and sends the remote system a signal to resume the transmission. ProTERM repeats this process as often as necessary until you either turn off the capture buffer or until you exit the program.

Using the ASCII protocol, again, makes ProTERM act exactly as if no protocol were used at all, except that all information is recorded.

When you are using the ASCII protocol to receive information and you do something that is not compatible with the protocol (like quitting the program or selecting another protocol), ProTERM automatically records the information currently in its internal workspace and shuts down the file.

Manually Saving the Buffer's Contents Even though ProTERM automatically saves all incoming information, you may wish to force all text in the internal workspace to be recorded or saved to disk. Press APPLE-S to save the buffer's contents.

Initializing the Buffer If information comes in that you don't want saved to disk, you can initialize or clear ProTERM's internal workspace and start fresh. Press APPLE-I to initialize the buffer. Since initializing the buffer is potentially destructive (to your data), ProTERM requires that you confirm the action by pressing the "Y" key. (This stands for "Yes.")

Turning off the Capture Buffer You can terminate an recording of incoming material by turning off ProTERM's capture buffer. Use APPLE-F to turn off the buffer. APPLE-F immediately turns off the buffer without saving what is in it, so if you want to save what is currently in the buffer before turning it off, use APPLE-S first.

ASCII VS Binary

The ASCII file transfer protocol is used for text. ProTERM lets you send programs, too. The other three protocols are used for this purpose. They are generally referred to as binary protocols because they are capable of transferring binary data. (You can use them for text, too, but their main use is for programs.)

When you get ready to receive a program, you must first inform the other computer to send the program. It will provide you with instructions on initiating this process. Once you tell the remote system to begin a file transfer (other than an ASCII transfer), it will wait for a signal from your Apple before it begins. ProTERM sends that signal once you've given it the name of the file in which the data is to be stored.

Summary of Program Transfer The following steps are used when receiving programs from a remote system using one of the binary protocols.

1. Give the remote system the command to send a file.
2. If necessary, tell the remote system the method for sending the file. This will usually be Xmodem, Modem7, or CRC.
3. When the remote system tells you to start the transfer on your system, give ProTERM the APPLE-R command.

4. When you press APPLE-R, select the file transfer protocol to use and give your new file a name.

While using one of the binary protocols requires that you setup things on the remote system first, ASCII protocol makes no such requirement. You can select the ASCII protocol then tell the remote system to send the text.

XModem

XModem is the normal XModem transfer protocol. Its advantage over the ASCII protocol is two-fold. First, it permits sending more than just text. When you send text, several characters are used for control purposes. Two of these, the CTRL-Q and the CTRL-S, are used to start and stop the transmission. That alone precludes these two characters from ever being transmitted. When you transfer a program, it might use those characters. Even if it doesn't directly use them, it probably contains codes that correspond to those two characters. XModem uses a different form of flow control. Instead of sending a stream of characters with control characters to start and stop the transmission, XModem sends its information in packets or blocks. The general size of one of these blocks is 128 bytes or characters. After a block is received, the local system tucks it away and sends the remote computer a signal that tells it that it was either good or bad. If it was bad, the remote system resends the block. This procedure is repeated until the entire file is transferred.

The way your program determines if a block was good or bad exemplifies the second advantage of using the XModem protocol. In addition to the 128 characters of data transmitted with each block, an additional byte is sent. It contains the sum of all the bytes of data that were sent in the block. (That's why this protocol is called checksum.) If this byte that contains the sum of all the bytes of data matches with the sum that ProTERM calculated while it received the bytes, ProTERM sends the remote system a signal that says, "OK, this block is good, send the next one." If ProTERM's checksum doesn't match with the checksum byte sent by the remote system, ProTERM sends the remote computer a different signal which says, "My checksum doesn't agree, send that block again."

There are two reasons that blocks of data might get rejected. First is the possibility of noise on the phone line changing one of the characters. Second, the checksum itself might get corrupted by noise. In either case, the data is sent again.

CRC ProTERM's implementation of the XModem protocol tries to use the more extensive CRC error correction technique. CRC is an acronym for cyclic redundancy checksum. It combines the simplicity of the standard XModem checksum technique with an algorithm that improves the error detection capabilities to well over 99%

accuracy. If, however, the other computer doesn't support this scheme, ProTERM automatically switches to the older checksum method.

The 1K Option ProTERM supports the transmission and reception of 1K blocks in the XModem file transfer. It automatically detects the 1K block size as it is sent and automatically reverts back to the standard 128 byte block size if the remote system rejects the 1K block or if several 1K blocks are rejected during a transmission. To send using the 1K block option, begin the transfer normally. Then, once the data starts, press the B key.

Binary II

Binary II is a special form of the XModem protocol. It lets you send several related files at a time. It does so by combining all the files into a single large file with a special format. What is more important, though, is that when you use the Binary II format, the statistics about the individual files in the larger, packed file are preserved. ProTERM recognizes this special format and automatically returns the included files back to their original state.

In order to use Binary II, the file to receive must have been packed in the Binary II format. The remote computer uses normal XModem file transfer protocol to send the file; Binary II's distinction is in the way the file itself is formatted. You can usually tell if the file is stored in Binary II format if its name ends with .BNY. Files can also have the extension .BQY. This is a special form of Binary II where some elements of the file are squeezed to save space and download time. If you aren't sure if a file is stored in the Binary II format, use the XModem protocol to download the file; ProTERM lets you unpack the file once it is stored on your disk.

If you select Binary II, ProTERM asks for the name of a directory where it will store the files as it unpacks them. If there isn't already a directory with the name you give, ProTERM creates one. The file you are downloading, of course, must have been packed with the Binary II format. If it wasn't, ProTERM asks you to give the file another name when it realizes that it isn't Binary II format. If the file is, indeed packed with the Binary II format, the files are unpacked and stored in that directory. If there are already files with the same name as the files being unpacked, ProTERM renames the already existing file so that its name is different. It does this by changing the last letter of the file's name to "A".

If you download a file packed in the Binary II format and you don't use the automatic unpacking capability provided with the Binary II choice from the Receive menu, you can still unpack the file with the APPLE-Z command. When you do, ProTERM asks, "Pack or Unpack". Press "U" to unpack the file then give ProTERM the name of the

directory where you want the files stored. (The "P" option has not yet been implemented on this version of ProTERM. If you want to pack several files into one file in the Binary II format, use the BINARY.UP program available from CompuServe, GENIE, or from the BAUD library.)

ProDOS

The ProDOS choice on the RECEIVE Menu refers to the AE Pro format which is a special form of XModem that includes file storage and date information. Its primary use is for transfers of single files between systems that support the AE ProDOS format. The information pertaining to the file's type and date information is gathered by the sending program and passed to the receiver at the time of the transfer. You won't, therefore, see AE ProDOS format used on the major information services like CompuServe and GENIE. Again, this protocol works only between two systems that support it. ProTERM is forgiving, however, if it sees that the other computer doesn't support the ProDOS format, it automatically switches to normal XModem format.

TRANSMITTING FILES

Transmitting files works much like receiving files. The direction, of course, is opposite.

Once you've instructed the remote system you are about to send it a file, press APPLE-T to access the transmit files menu. This menu looks exactly like the receive files menu. Again, use the ARROW keys or the first letter of your choice to select the protocol to use, then press RETURN. ProTERM asks for the name of the file to send.

Cancelling a File Transfer

During the transmission of a file, you may decide that you no longer wish to send the remainder of the file. You may press the ESCAPE key during a transmission to cancel it. When you press ESCAPE while transmitting using the ASCII protocol, ProTERM quits sending data, and returns to terminal mode. If you are using one of the three binary protocols, ProTERM asks, "Abort?". Press Y if you really want to stop. If you do, ProTERM informs the remote system and goes back to terminal mode.

Altering a Filetype

ProTERM permits you to alter the file type of any file on disk. This is an extremely powerful and dangerous feature. If you don't know why you would want to alter a file's characteristics, don't do it.

When you are ready to alter the file characteristics, you should press APPLE-A. APPLE-A accesses the "Adjust Filetype" section of the program. This process allows you to alter any filetype to any other filetype.

When you press APPLE-A, ProTERM asks for the name of the file to alter. Type the file's file or path name and press RETURN. ProTERM next shows you a menu that shows the current filetype. Use the ARROW keys to move the pointer to the desired filetype. You can move ten filetypes at a time by holding down the APPLE key while pressing the ARROW. When the pointer reaches the filetype to which you wish to change the file, press RETURN. Next, ProTERM allows you to alter the auxtype of the file. The auxtype is used for a variety of purposes depending on the type of the file. If the file is a text file, set it to zero. If it is a BASIC program, set it to \$801. If it is a type SYS file, set it to \$2000. If the file is binary, you need to know the loading address of the file in order to use it.

Remember, if you are using ProTERM to communicate with another ProTERM user or any other system that supports the extended XModem protocol or if you use the Binary II protocol, you don't have to worry about the APPLE-A command.

Macros

A macro is a group of characters that is automatically typed in for you when you type a single letter. This section of the manual describes ProTERM's support of macros, explains the structure of macro files and definitions, and outlines the various macro functions.

Structure of a Macro File

A Macro file is an ordinary text file that contains definitions of commands and keystrokes that are to be entered when you press a single command. When ProTERM first starts, it automatically loads a macro file from your disk. That file appears in the same directory where ProTERM itself is located and is called TERM.MACROS.

TERM.MACROS contains two things. First, it contains a single identification character at the beginning of the file. That character is CTRL-A. If the file doesn't begin with CTRL-A, ProTERM won't use it as a macro file. The CTRL-A appears only at the beginning of the file, not at the beginning of each macro definition. Secondly, TERM.MACROS contains a series of macro definitions.

Structure of a Macro Definition

A macro definition consists of four parts. They are (1) the macro access character, (2) a delimiter character, (3) the macro keystrokes and commands, and (4) the final delimiter character.

The Macro Access Character The macro access character is a single letter, number, or punctuation symbol that, when pressed with the OPTION key, makes your macro spring into action. Every macro definition begins with the access character.

The Delimiter The delimiter character is any character that won't be used within the command and action part of your macro. It is used merely to let the program know when the end of the macro comes.

The Macro Action and Command Section The macro action and command part of the macro is the "meat" of the macro. Whatever appears between the two delimiter characters is what the macro does when you press the macro command character.

The Final Delimiter The final delimiter is the same character used as the initial delimiter character that follows the macro access character. It is used to signal the end of the macro. The next macro's definition begins immediately after this macro's final delimiter.

Example of a Simple Macro

The macro shown below dials the number 555-1212. It, like every macro, begins with the macro access character, contains a beginning delimiter, contains the modem command to dial the phone, the number to dial, and a carriage return, and the macro ends with the final delimiter. The macro access character in this example is D. The delimiter is @.

```
C@ATD555-1212
@
```

Notice that the second delimiter appears on the second line of the macro definition. That's because the macro contains a carriage return after the final digit in the phone number.

The next macro access character (for the next macro) always appears immediately after the final delimiter from the previous macro. If a second macro existed in this file, the file would look like this:

```
C@ATD555-1212
@E@ATD555-1213
@
```


The Macro Command Language

Besides using macros to type straight text, you can use macros to issue commands to ProTERM. Anything you can do from the keyboard, in fact, can be done with a macro. There are, additionally, several features you can use from macros that you can't do from the keyboard.

Using APPLE and OPTION Commands in Macros As discussed, macro files are simple ASCII text files. There is, however, no way to represent APPLE and OPTION using ASCII. It is for that reason that a code has been designated for this purpose. The code consists of three parts. It begins with the CTRL-O character, is followed by either 0, 1, or 2, and concludes with the letter pressed with the APPLE or OPTION key. The numbers 0, 1, and 2 that follow the CTRL-O represent the APPLE, OPTION, and both the APPLE and OPTION keys respectively.

Example of a Macro Using the APPLE Key If you want a macro to receive a file, it might look like the example below. This macro's access character is R. It starts receiving an ASCII file and gives that file the name /DATA/FILE. CTRL characters are represented in these macros by preceding the letter with the ^ symbol.

```
R@^OORA  
/DATA/FILE  
@
```

Notice that this macro contains every keystroke necessary to do this from the keyboard. First, an APPLE-R is pressed. That is represented with the CTRL-O, the 0, and the R. Next, the letter A is pressed. That moves the selection bar to ASCII on the menu. After the A, the macro contains a carriage return to select ASCII. Next, the name of the file is typed and a carriage return is pressed.

If you wanted to select 1200 baud, set Half Duplex to "Yes", and make sure that the information clicker is set to "Yes" before the APPLE-R command is used, the macro would look like this:

```
R@^OOPB1HYIY  
^OORA  
/DATA/FILE  
@
```

Note that this macro now begins with CTRL-O, 0, P which is the same as pressing APPLE-P. Next, the macro presses B to move the selection bar to the baud rate option on the parameters menu. It then presses 1 to select 1200 baud. Next, the macro presses H to move the selection bar to Half Duplex and presses Y for "Yes." Finally, the macro presses I to move the information bar to incoming data clicker on the parameters menu, presses Y for "Yes"

and presses RETURN to exit the menu. The rest of the macro receives an ASCII file as described previously.

Calling Other Macros from within a Macro One of the most useful features about ProTERM's macro command language is the ability to call other macros from within a macro. The command character to call other macros is CTRL-Y. The CTRL-Y macro command contains three parts. First is the command character, CTRL-Y. Next comes the letter describing the macro to call. Finally, the CTRL-Y command is terminated with a carriage return. You can include notes or comments after the macro letter to call and before the carriage return.

The CTRL-Y macro language command is much like calling a sub routine. That means that when the macro that is called comes to an end, execution of the macro that called the other macro resumes immediately after the CTRL-Y command.

There are two macros included on your ProTERM disk that come in useful as parts of other macros. They are the] and the % macros. The] macro is used to establish a connection with a remote computer. It continues redialing the phone until the other computer answers. It assumes that you have a smart modem that responds with the word "CONNECT" when connection is established. It is designed to be used after using the modem's dial command to dial a phone number. The second included macro, the %, is used to pause while a series of common prompts is received.

The macro previously described that dials 555-1212 looks like this when you use the] macro to dial until a connection is established.

```
C@ATD 555-1212  
^Y]  
@
```

This macro does the same as the first macro--dial the number--and waits for the modem to send the CONNECT message. If the modem doesn't send that message, the] macro continues re-dialing the number until it does. You'll want to use the] macro for all your dialing.

Waiting for a Series of Prompts and Finishing the Waiting There are two macro language commands that let you wait for a series of prompts and a macro command that finishes the waiting sequence. They are CTRL-W and CTRL-F respectively. They are always used together.

Format of Waiting The wait and finish waiting macro commands consist of four distinct parts. They are (1) the initial CTRL-W command, (2) the text to wait for, (3) the place to go if this text is found, and (4) the final CTRL-F to finish the sequence. (It is

always a good idea, to improve readability, to follow the CTRL-F with a carriage return.) You separate each of the parts with a delimiter, ending the final search sequence with two delimiters. This scheme sets up the program to wait for one of the phrases. The CTRL-F puts the macro language into a suspended state so that it does nothing else until one of the phrases is found.

Example of a Wait Sequence The following macro waits for the words "CONNECT", "BUSY", "NO CARRIER", "VOICE", and "NO DIAL TONE". The delimiter used in this example is the carriage return. If the characters for which you wait don't include a carriage return, it is a good idea to use the carriage return as the delimiter, it improves readability of the macro file tremendously.

```
^B0
^IA/
^P Re-dialing
^B2
^W
CONNECT
8
BUSY
1
NO CARRIER
1
VOICE
3
NO DIAL TONE
3

^F
^B3
^OP Busy or some one using the phone
^P Hanging up

^B9
^B8
^P Connected
^B9
```

Branch Points In conjunction with the CTRL-W, wait sequence command, are associated a number of branch points. These are labels that identify a part of the macro as a place for the macro to resume if one of the phrases is found.

Branch point labels have three parts. They are (1) CTRL-B to mark this as a branch point, (2) a number from 0 to 9 specifying the name of the branch point, and (3) a carriage return. The carriage return is required (and includes readability), but you may include any comments or notes between the branch label and the carriage return. A branch point looks like this:

^B1

^B1 This is branch point 1 with a comment.

Branch points are used for two things. First, as you've seen, they work in conjunction with the waiting sequence command. Secondly, they are used with the CTRL-J jump to branch point, command.

Jump to Branch Point CTRL-J is used to command a macro to skip a section of the macro and resume at the indicated branch point. It has three parts. They are (1) CTRL-J command character, (2) the label where to jump, and (3) a carriage return. Again, you may include notes or comments between the label and the carriage return.

Printing Messages on the Screen CTRL-P is used to make a macro put messages on the screen. The CTRL-P command has three parts. They are (1) the CTRL-P command character, (2) the message, and (3) a carriage return.

Slight Pause The macro command language includes a command to make your macro pause for about one second. This is sometimes useful to wait for slow responding computers. The command character is CTRL-I. The CTRL-I command is used in the] macro to make the macro wait before re-dialing the phone in case you have one of the services that works by quickly clicking the plunger on the phone. If the macro didn't pause after hanging up the phone, it is possible to access the second line when the macro re-dials.

Combining the Parts Now that you know about CTRL-W, CTRL-F and CTRL-B, a look at the supplied] macro should prove useful. The macro looks like this:

Command	Comments
]@^P Dialing...	Access character, delimiter, and print "Dialing"
^J2	Jump to branch point 2
^B0	This is branch point 0
^P Re-dialing	print "Re-dialing"
^Ia/^B2 ^I	pauses, A/ redials, ^B2 marks branch point 2
^W	Begin wait sequence with RETURN as first delimiter
BUSY	wait for "BUSY"
0	Go to branch 0 if found (to re-dial)
CONNECT	Wait for "CONNECT"
1	Go to point 1 if found
VOICE	Wait for "VOICE"
3	Go to point 3 if found
NO CARRIER	Wait for "NO CARRIER"
0	Go to point 0 if found
	Second carriage return delimiter to end waiting

```

^F                ^F to suspend macros while waiting
^B3               Branch point 3
^P Someone's using the phone! Print message
                  Carriage return
^P Hanging up the phone...   Another message
^J9               Jump to branch 9
^B1               Branch point 1
^P Connection established!   Print the message
^B9               Branch point 9
@                 End macro

```

As the comments to this macro indicate, it begins by printing the word "dialing". This is done, again, with the CTRL-P command. Next, the macro skips down to branch point 2 with the CTRL-J, jump, command. At branch point 2, a wait sequence is initiated with the CTRL-W. Immediately following the CTRL-W is a carriage return which is used as the delimiter. (Using the carriage return as the delimiter in a wait sequence dramatically improves its readability, but, of course, can't be used if you are actually looking for a carriage return; remember, the delimiter must be a character not used in the waiting sequence.) After the initial wait sequence delimiter, comes the first phrase to look for followed by the delimiter and the branch point label where the macro should go if this phrase is seen.

The first phrase is "BUSY". If that phrase is seen, the macro goes back to branch point 0 where the CTRL-I, pause, command is issued followed by the modem's re-dial command, the A/. Immediately following the modem's re-dial command is the branch point 2 label, the place the macro began the wait sequence that caused it to go back to re-dial when "BUSY" was seen.

After "BUSY", the carriage return delimiter, the 0, and another carriage return delimiter, comes the next phrase to wait for. It is "CONNECT". After the word "CONNECT", comes the carriage return delimiter, the digit 1, and another carriage return delimiter. Look down to where branch point 1 is located in the macro (the ^B1), and you'll see that the macro prints "Connection established" and then comes the end of the macro. The rest of the waiting sequence follows this pattern. If "VOICE" is seen, the macro prints a message that a person answered the phone and exits. If "NO CARRIER" is found, the macro branches back up to branch point 0 and the modem is re-dialed.

Another Waiting Sequence The following macro (also included on your disk) shows another example of a waiting sequence. This one, however, waits for a carriage return as part of one of the phrases, so the carriage return can't be used as the delimiter. Instead, / is used.

```

%@^B0 Waits for "!" "OK" or "orum" as CompuServe prompts
^W/ore!/9/

```

```
OK/9/unction:/9/
!/9//^F
^B9
@
```

As you see from the first character of this macro, it is accessed with the % key. Its purpose is to wait for one of the CompuServe prompts. If you design macros to move through specific services in CompuServe, you'll want to use the % macro to wait for a prompt before continuing with the next CompuServe command.

The reason the carriage return couldn't be used in this waiting sequence is that it is used to check for the prompt "OK". It wouldn't do to wait for OK by itself because OK can appear within a word. The chance of a carriage return followed by OK, however, is slim.

CompuServe Logon Macro The macro listed below, with a little modification, logs you on to your local CompuServe network. There are three lines you need to change to make this macro work for you. They are the ones that dial the number, type your ID, and type your password. The lines are clearly labeled.

```
c@
^P CompuServe
^O0DCompu
^[^O0ra
Compu
Replace this line with "ATD" and the phone number
^Y]
^B2
^C^W
ID:
5

^F
^B5
Replace this line with your ID
^W
Password:
6

^F
^B6
Replace this line with your password
@
```

The CompuServe logon macro does several things. First, it deletes a file named COMPU in the open folder. Next, it prepares to receive a file named COMPU in the open folder. The macro deletes the file because, if you don't, ProTERM continues adding to the end of the currently existing file when using the ASCII protocol. You

may want to remove the line that deletes the file if you feel uncomfortable with this. You may also wish to add a volume name to the name of the file to put it on a specific disk no matter the open folder.

The next line in the macro is the one that you should replace with "ATD" and the local CompuServe access number. After the line that dials the number, is a call to the] macro which, again, keeps trying to dial the phone until a connection is established. When a connection is established, the macro sends a CTRL-C. That gets you directly into CompuServe. Next, the macro starts a waiting sequence that looks for the phrase "ID:". The line that instructs you to replace it with your ID will be sent to CompuServe when ID: is seen. Finally, the waiting sequence is repeated, this time waiting for "Password:". Again, replace the line that indicates you do so with your password.

The Supplied Macros The table below lists the supplied macros.

] dials a number.
% waits for CompuServe to give you a prompt.
' hangs up the phone (if you have a smart modem with support for extended commands)
\ prepares the modem and program for an incoming call. (It also depends on a modem that supports extended mode to adjust your baud rate to the incoming caller's.)
& Resets the modem to no answering, touch tone dialing, and extended command mode.

In addition to these commonly used macros, ProTERM reserves several other macro characters. They are the digits 0-9 and the punctuation COMMA, DASH, PERIOD, and PLUS. These are reserved for use with VT100 emulation. If you don't ever use VT100 emulation, you can remove those macros from the table and substitute your own for those letters.

Monitoring a Macro

When a macro is issued, the keystrokes it makes and the results it produces are not spoken. Most of the time, you don't want to hear them. There are times, however, especially when developing a new macro, that you want to know exactly what is going on. You can monitor a macro's progress by pressing the APPLE key. You can also use the macro language's CTRL-P, print, command to place messages at strategic points in the macro.

Issuing a Macro

Once you've written a new macro, hold down the OPTION key while you press the macro access character to make that macro go. If your macro access character were D, you would press OPTION-D to make it work.

Stopping a Macro

There are times, most notably while under development, that a macro needs stopping. You can halt a macro dead in its tracks with a press of the ESCAPE key.

PART V: ProTERM Reference

Key Function

CTRL-D dumps the speech buffer.
CTRL-E adjusts TEXTALKER and ProSCAT speech.
CTRL-L enters review on TEXTALKER.
CTRL-O lets you send other CTRL chars.
CTRL-Q restarts the remote system.
CTRL-R enters review with ProSCAT.
CTRL-S stops the remote system. (See also CTRL-APPLE-a.)
CTRL-X silences speech.

APPLE-A alters a file's type.
APPLE-CTRL-A sends the stop signal to a remote system.
APPLE-C copies a file.
APPLE-D deletes a file.
APPLE-E executes a program.
APPLE-F turns off the capture buffer.
APPLE-I initializes the capture buffer.
APPLE-L loads a macro file.
APPLE-M makes a directory.
APPLE-N changes the name of a file, folder, or disk.
APPLE-O lets you specify characters to omit.
APPLE-CTRL-O cancels omitted characters.
APPLE-P sets parameters like baud rate, data bits, etc.
APPLE-Q quits.
APPLE-R receives data to a disk file.
APPLE-S saves the capture buffer.
APPLE-T transmits a file.
APPLE-U unlocks and locks files.
APPLE-CTRL-U sends CTRL-U to the remote system.
APPLE-V lets you view the contents of a file.
APPLE-UP ARROW scrolls back through the buffer.
APPLE-COMMA opens a new folder.
APPLE-QUESTION MARK shows a list of files on the open folder.
APPLE-SLASH shows files in the main folder of the last disk drive accessed.
APPLE-PERIOD shows disks in all your drives.
APPLE-LEFT BRACKET saves ProTERM's defaults.
APPLE-LEFT lowers the volume on a //gs.
APPLE-RIGHT raises the volume on a //gs.

A letter typed with the OPTION key executes the macro assigned to that key.

PART VI: Appendixes

Appendix A: Background

This appendix provides new and prospective computer users with some information they should know. It describes some basic functions of the personal computer and some details on what it is and how it works. The examples presented here are specific to the Apple computer but the general concepts apply universally.

Appliances

When you want toast, you pop your bread in the toaster. To open a can, you grab the can opener; coffee, the coffee pot. Modern man employs special appliances for every job. The personal computer is an appliance too. But, unlike can openers and coffee pots, the computer is programmable. This means you buy costly machinery only once. When you desire a new appliance, you write or buy a set of instructions that transforms the computer into the tool you need.

Programs

A program (software) is a group of instructions that makes the computer do its job. Programmers write the instructions and store them on a floppy disk. There are programs for nearly every job imaginable. A program might, for instance, manage your checkbook by accepting transaction figures and printing checks and reports. Another takes the knighted user through haunted castles and imaginary universes. The most popular programs for home users are applications; that is software that does a job. Most personal computerists use a word processor, data base, and telecommunications software.

Getting a Program Running

To get a program started, you insert the disk with the desired program on it into the disk drive, close the drive door, and turn on your machine. The program then takes control of the computer. It displays instructions or gets itself ready to accept your commands.

Floppy Disks

A floppy disk is a median of storage for either programs or data. It works much like a normal cassette tape--storing by means of magnetic patterns. Unlike a cassette tape, however, the disk may be accessed randomly. In other words, you need not move the tape past all the unimportant information to get to the place you need to be.

Why use Floppies The computer's memory is "volatile". When you shut off the power to the machine, any programs or work you had generated with those programs disappears. That is why you use disks to store your work before shutting off the machine. In earlier days, cassette tapes stored most programs and data but the data transfer rate on disks is much faster. Another reason for the shift to disks is their "turnkey" operation. In other words, the user can simply insert the disk and begin using the machine without worrying about loading individual programs and data files from a cassette machine.

Many programs allow you to create large documents or lists of names and addresses. You use disks to store this information in order to continue using the same data from session to session.

Commanding a Program

When you are ready to use some previously generated data or when you want to record your work on disk, you give the program a command. you find out what command a program expects for each task by reading that program's manual or by reading instructions displayed on the screen. Nearly every program differs in the commands it uses. Fortunately, however, some commands, especially of the editing type discussed below, have become standards employed by many programs.

Some programs are menu driven. Like the menu in a restaurant, the program's menu offers a variety of options. You pick the option, in many menu-driven programs, by typing the number displayed next to the item on the screen. Other programs using menus permit choosing by typing the first letter of the desired item. Still others let you move a pointer up and down the screen to make your choice. Most menu-driven programs display specific instructions on how to make a selection.

New users enjoy menu-driven programs. They always have a complete list of options. After gaining experience with the program, however, the menu slows down your effective use of the computer. You may often, for instance, choose an item on the menu only to access yet another list of choices. That list may lead to yet more lists. This may continue until you are several layers deep into the menus. You must then thread your way back out. By the time you go through all that, chances are that you've forgotten what it was you wanted to do in the first place.

Menu-driven programs are not the only way to issue commands. Many of the more-complex programs use a command line to accept commands. You, in these programs, type an English-like command to instruct the computer about what you want to do. This permits both a wider variety of commands and immediate access to all of them. When a program uses this method of accepting commands, you must usually press the RETURN key to inform the program that you are through

typing and ready to execute the command. Both the Apple's operating system and BASIC language employ this method of accepting commands. The DOS command "CATALOG", for instance, shows you a list of files on the disk. If you type "RUN", BASIC interprets it as the command to execute the BASIC program currently in memory. (BASIC stands for Beginner's All-purpose Symbolic Instruction Code.)

Besides menu-driven and command line methods of executing commands, the command key offers a method employed mainly by programs that let you edit what you type. A command key might be the BACKSPACE which backs up the cursor and lets you type another character in its place. The command key method of executing commands is popular in editing software because it takes only one quick keystroke to perform a task that you might need to use frequently.

The Keyboard

Notice the similarities between the computer's keyboard and a typewriter's. Both contain enough keys to type every letter, number, and many punctuation marks. Note, too, the differences. The computer includes several keys - ESC, DEL, and CTRL - not found on typewriters. They perform special tasks. The CONTROL key, for instance, works much like a SHIFT key. When held down while another key is pressed, a new character is formed. Often these control characters are used as command keys.

Program Interaction

When you turn on your machine, a complex series of programs begins. There is a part of the computer's memory called "ROM" (read-only memory) which does not change even when power is interrupted. As soon as you turn on your machine, one of the programs stored in ROM begins initializing the machine. One of the first things this program does is scan the slot connectors on an Apple. (Apples have seven connectors under the cover in which one may insert peripheral cards. These cards connect external devices such as printers, modems, and disk drives to the computer.) If there is a disk controller card in one of the Apple's slots, yet another program, contained in the ROM on the disk controller card, begins loading the operating system from the disk in the drive connected to that card into memory.

A disk operating system (DOS) is a complex program which performs the transfer of data between the computer and the disk. When the program on the disk controller card's ROM first starts, it loads a small section of the actual operating system from the disk. This small section of code is generally a set of subroutines that load more sections of the program. (A subroutine is a group of general instructions that other programs may use.) This process continues until the complete operating system is loaded into memory and moved to its executing address. This complex series of steps is called

"booting" because the operating system pulls itself up by its own bootstraps.

When the operating system loads on an Apple, it tries to load and execute a BASIC program with a certain name. If there is no such program or when the program ends, you see a right square bracket appear on the left side of the screen. This is the BASIC language command interpreter's prompt character. The BASIC language is still another program that runs in conjunction with DOS. BASIC is a high-level language which accepts English-like commands so that even computer novices may write their own programs. It is stored, for the most part, in ROM on the Apple and is called Applesoft. When it shows its prompt character, it is waiting for you to type a command.

When you type a command on the Apple computer, the operating system first looks at the command to see if it can execute it. If DOS does not recognize the command, the BASIC language looks at what was typed to see if it can understand it. If Applesoft does not know the command either, it makes a harsh beep and shows you an error message like "?SYNTAX ERROR". If, on the other hand, the line that you type begins with a number, Applesoft assumes that this line is part of a program and stores that line into memory adding it to whatever other lines are present. (When you give the command to execute the program, Applesoft begins performing the instructions contained in each line of the program beginning with the lowest numbered line.)

When you see the BASIC prompt character, you may enter DOS commands, BASIC commands, or BASIC program lines. It is important to know the difference between DOS commands and BASIC commands. If you don't plan on programming in BASIC, your primary interest will be DOS commands. DOS commands are statements that, for the most part, may be typed either directly from the keyboard or used from within a BASIC program. These commands perform essential file manipulation operations such as loading files from the disk into the computer's main memory, saving files from memory to disk, and general housekeeping functions such as deleting files from the disk or just showing what files exist on any particular disk. (A file is a group of related information which is either a program or data that is stored under one name on the disk.)

Machine Interaction

While the computer's programmability makes it a tool unparalleled in history, it is its ability to communicate with a wide variety of devices that makes it such a flexible tool. Handicapped users especially benefit from the ability to utilize the computer's processing power to communicate with relatively-inexpensive peripherals such as speech synthesizers, braille printers, and optical-character scanners.

The computer consists of three main parts: the input section, usually a keyboard; the processing section; and the output section, usually the screen. The input section is used to provide the processor with information and instructions. The processor does not care where this information comes from. The output section tells the computer's operator the results of the information or commands it has processed. Again the processor does not care whether this output goes to a screen, a printer, or a speech synthesizer.

Most blind Apple computer users choose the Echo speech synthesizer, from Street Electronics, as their primary form of output. The Echo, with its accompanying software, provides blind users a way of sending everything that is printed on the screen to the speech synthesizer. This allows them to use many of the programs available to the general public. More importantly, this system provides some special features designed especially for the blind.

While any synthesizer performs essentially the same job, it is the Echo's accompanying software and the synthesizer's low price that make this particular system so popular to the blind. The software provided with the Echo synthesizer, a program called TEXTALKER, enhances the Echo's utility in several significant ways.

TEXTALKER voices your keystrokes. If you type a capital letter, the pronunciation of that letter occurs in a slightly-elevated pitch. More importantly, you may enter a special "review" mode which permits you to review any material currently displayed on the screen.

One could easily think that, since the program sends all its output to the speech synthesizer, all is well for the blind user. This is not always true. Many programs, for instance, decorate the screen with hundreds of "*"s. These ornamental touches please sighted users but many blind users cringe and hold their breath when they begin hearing the seemingly endless series, "star star star...". The Echo user may, however, press the CTRL-X to temporarily silence the speech, skipping this unwanted material.

Other programs may repeatedly display all the instructions after every action. This again causes needless delays to the blind user.

Some software is designed with the blind in mind. Such software considers the affects of speech on the material being presented. Specially-designed talking software uses speech to both enhance the learning experience and to increase the utility of the software. By carefully designing a program, the programmer creating talking software provides the blind user with material that enhances his experience on the computer and provides an important means of performing those tasks that before were nearly impossible. Look, for instance, at the importance of a good talking word processor to the blind user. Without this tool, his writing techniques consist

of either taping or brailling a rough draft of the text. Next, he relies on the typewriter which provides no feedback on where he is on the page or what was the last character typed. (It is not difficult to see that one would have problems trying to remember if he had already spaced after that last sentence.) The word processor provides the blind user a tool to use from the creating stage of writing right through to the final copy. The ability, alone, to read and manipulate his text proves most useful. When one adds the fact that the word processor always lets the blind user know exactly what he has typed and where he is on the page, it is not difficult to see the value of such software to the blind.

Appendix B: Utilities

Two programs included on the master program disk--FILER and CONVERT from Apple Computer, Inc.--have been modified to work with speech. Each is a utility for use in the ProDOS operating system. Each is menu driven and provides the new user with help by pressing the "?" from any menu. Once a menu item is selected, you may back out to the previous menu by pressing the ESC key at the upper left of the keyboard.

Filer

FILER is probably the more-frequently used of these two valuable programs. Unlike Apple's earlier operating system, DOS 3.3, ProDOS contains no command to format a disk. There are several reasons for this omission--lack of space for the code and ProDOS's flexibility over a wide range of storage devices to name a few. To format a disk under the ProDOS operating system, you must use the FILER program.

FILER provides much more than just formatting disks; you may use it to copy files and disks, delete files, or, to name a few, compare files. When FILER is run, a main menu appears. This menu lets you work with either entire disks or individual files.

Working With Disks (Volumes) A volume is the entire storage medium. The volume might be a 5.25 inch floppy disk, a 3.5 inch disk, a RAM volume, or a mass storage device--like the ProFILE or SIDER hard disk systems. If you select volume commands (V) from FILER's main menu, another menu appears. This is the volume menu and it contains several more choices. The two we are interested in are copy a volume (C) and format a volume (F). Remember, when copying an entire volume, FILER automatically formats the new volume so it is not necessary to format a volume before making a copy onto it. If, however, you wish to copy individual files to another volume, the new volume must already be formatted.

Copying Volumes If you choose "C" (for copy a volume), FILER requests that you supply the slot and drive numbers for the original volume and the slot and drive numbers of the volume on which the original will be copied. (The slot number, usually six, refers to the expansion slot in which the disk controller card is connected and the drive, either 1 or 2, refers to the disk on that card.) FILER supplies you with defaults which may be set using the set defaults selection (D) from the main menu. You may accept the default by pressing RETURN or you may type another number. (It is not necessary to press RETURN after typing the number.)

To copy the contents of an entire volume from the disk in drive one to the disk in drive two, the numbers to press would be 6, 1, 6, and 2. If you have only one disk drive, you would type 6, 1, 6,

and 1 again. FILER prompts you to swap the disks as necessary when using a one-drive system. Once the appropriate numbers are typed, FILER asks you to insert the disks and then press RETURN to let FILER know that you are ready to proceed.

When you press RETURN, FILER looks at the original volume and shows you the name of that volume. FILER asks you to either press RETURN to give the copy that same name or to type a different name to assign to the new volume.

If the disk on which the original is being copied has already been formatted, FILER shows you the name of that volume and asks if it is alright to destroy it. This prevents you from accidentally destroying possibly valuable data on a disk. You may press "Y" to proceed or "N" to remove that formatted disk and start the process again.

Once the volume is copied, FILER returns to the menu that allows you to enter the slot and drive numbers of the original and duplicate volumes. Remember, pressing ESC backs out one menu level.

Formatting Volumes Choosing "F" (for format a volume) from FILER's volume menu allows you to format your disks. (Every disk must be formatted before you can use it to store data.) The "F" selection permits you to enter the slot and drive numbers of the disk drive in which the volume to format is located. Again, FILER supplies defaults which may be accepted by pressing RETURN or overridden by typing the appropriate slot and drive number.

Once the slot and drive numbers are entered, FILER supplies a default volume name. This name is BLANK00. You may press RETURN to give the new volume this default name or you may type another name. Usually, you want to assign your disks more meaningful names; JUNK, NOTES, HOMEWORK, UTILITIES, and FINANCIAL are examples of volume names you might use.

If the disk that is about to be formatted has already been formatted, FILER gives you a chance to back out by showing you the name of the volume and asking if it is alright to destroy the contents of that volume. Again, you may press "Y" to proceed or "N" to start over. Since ProDOS contains no internal-formatting capability, it is a good idea to keep several formatted volumes on hand. Once you have formatted a few disks, you may press ESC until you back out to the main menu.

Working With Files On the main menu, "F" selects file commands. The file commands menu contains several selections. When using the file commands section of FILER, the program requests the complete pathname of the affected files. No slot and drive parameters are permitted. You may, however, include wild card characters in the pathnames.

Wild Cards There are two wild card characters, the QUESTION MARK and the EQUALS SIGN. When the QUESTION MARK is used, FILER asks you to confirm each file that matches the other parts of the pathname specified. The EQUALS SIGN wild card indicates to FILER that it should perform the specified action without prompting you for confirmation.

Copying Files If you select "C" (for copy files) from the file commands menu, FILER asks you to type both the name of the file to copy and the name of where the file should be copied. If you have a disk named JUNK and there were a file named MISC on that disk that you want copied to the disk named NOTES, the names to type would be

/JUNK/MISC

/NOTES/MISC

You could make FILER show you the names of each of the files on the JUNK disk by typing

/JUNK/?

/MISC/?

You could make FILER copy all the files on the JUNK disk to the MISC disk without asking you by using these names

/JUNK/=

/MISC/=

If there are any files on the MISC disk that already exist with the name of one of the files you copy, FILER pauses and requests permission before replacing that file with the new information from the JUNK disk.

Copying to the RAM disk You may use the RAM volume to store files or programs that are often accessed. Many users like to copy WORDS, TERMINAL, and FILER to the RAM volume. You could copy the files you want by using the following names

/USER/?

/RAM/?

This makes FILER look at each file on the USER disk and ask for your permission to copy it to the RAM disk.

Convert

The CONVERT program lets you transfer files between the DOS 3.3 operating system and the ProDOS operating system. This program is not well adapted for speech. The American Printing House for the Blind, however, sells a disk containing both FILER and CONVERT in versions that have been completely re-written for speech. The disk costs about \$20 and comes with instructions on tape. You can contact APH at (502) 895-2405.

Not all programs will work in an operating system other than the one under which they were designed. Most BASIC programs and text files, however, transfer between the two operating systems with no problem.

CONVERT starts by assuming several things about the transfer of files. First, it seems to require two drives. Secondly, it assumes that you wish to transfer files from DOS 3.3 to ProDOS. These setup parameters are easily changed with one of the choices from the main menu.

The main menu offers several clearly titled options. The direction of transfer may be switched by selecting the reverse direction (R) choice on the main menu. "P" allows you to set the prefix for the transfer. Selecting "D" from the main menu displays the current date and permits the user with no clock card to enter the correct date. The default parameters for the DOS 3.3 disk are altered with the change DOS 3.3 slot and drive (C) option.

There are three steps involved in making a file transfer. One must, of course, first execute the CONVERT program. This may be done from ProWORDS or ProTERM by pressing APPLE-Q then typing "CONVERT" or "/USER/CONVERT" if the prefix is not "/USER". The CONVERT program needs to know where the DOS 3.3 disk is and where the files will be transferred. Tell CONVERT where the DOS 3.3 disk is by using the "C" selection on the main menu. (The "C" selects the "CHANGE DOS 3.3 SLOT AND DRIVE" option.) Once "C" is pressed, CONVERT asks for the number of the slot and drive where the DOS 3.3 disk is. It is not necessary to press RETURN after entering the slot and drive numbers. Normally, the disk drive's controller card is connected to slot 6.

Once the DOS 3.3 slot and drive is set, use the "SET PRODOS PREFIX" item by pressing "P". CONVERT asks you to choose the method of selecting the prefix, by slot and drive (S) or by pathname (P). If you choose to set the ProDOS prefix by pathname, CONVERT asks you to type the name of the directory to which the file or files will be transferred. If "S" is selected, CONVERT requests the number of the drive of the ProDOS disk. After either entering the pathname or the slot and drive, CONVERT returns you back to the main menu.

Transfers in the other direction, ProDOS to DOS 3.3, are done in the same way. One must, however, also reverse the direction of transfer by pressing "R" from the main menu.

Once the DOS 3.3 slot and drive and the ProDOS prefix are set, you are ready to transfer the files. Begin this procedure by pressing "T". When "T" is selected, CONVERT asks you to either type the name of the file to transfer or to press RETURN for a list of files -- If you don't have the special talking version, DO NOT DO IT! The speech synthesis user will find nothing but confusion and much, much talking. Instead, type the name of the file. An efficient means of transferring many files also exists. Simply make sure that the DOS 3.3 disk contains nothing but files that are to be converted to ProDOS. Next, after pressing "T" to transfer the files, type a EQUALS SIGN but don't press RETURN yet. Instead, type the command to silence the speech. (CTRL-E followed with "O".) After selecting TEXTALKER's screen only mode by typing the CTRL-E "O", press the RETURN key to begin the transfer. When the disk quits spinning, you may wish to re-select TEXTALKER's both screen and talk mode by typing CTRL-E "B". Press ESCAPE to return to CONVERT's main menu.

Converting files on one drive systems The ability to make a one-drive transfer is less obvious. One should, in short, utilize the /RAM volume as a temporary place to store the ProDOS file. If converting from ProDOS to DOS 3.3, first, using the FILER program, copy the file to /RAM then, after entering CONVERT, set the prefix with option "P" to /RAM. After that, just make sure that the slot and drive are set for the DOS 3.3 disk, press T to transfer, and type the name of the file.

Quit CONVERT Once the converting is complete, one may press "Q" from the main menu to quit the program. When "Q" is selected, CONVERT asks for the name of the next file to execute. Type the program's name and press return. (Note, the next program must be a system file.) To execute words, for instance, type "/USER/WORDS" and press RETURN. To execute terminal, type "/USER/TERMINAL" and press return. Similarly, to execute FILER, type "/USER/FILER" and press RETURN.

Appendix C: Pathnames and the Selector

When ProWORDS or ProTERM asks you to enter the pathname of a file (like when you use one of the APPLE commands), you can do more than just type the name of the file. Think, for instance, if you don't remember the name of the files on your disk. You can press the QUESTION MARK to see a list of them. This section of the program, called the selector, lets you use the UP and DOWN ARROW keys to point to a file. Once you point to that file, you press a command key to tell the selector what to do with that file. If, for example, there were a file on your disk named NOGOOD that you wanted deleted, you use the UP or DOWN ARROW to point to the file, then you press D to delete it.

Accessing the Selector

There are several ways to enter the selector. As you have seen, you may press QUESTION MARK at any "ENTER PATHNAME" prompt. In addition, you may press, at any time, APPLE-QUESTION MARK or APPLE-SLASH. (Pressing APPLE-QUESTION MARK shows you a list of files in the current folder; APPLE-SLASH shows a list of files on the main volume folder of the disk in the last drive used.)

Using Other Folders In addition to typing QUESTION MARK at the "ENTER PATHNAME" prompt or using APPLE-QUESTION MARK or APPLE-SLASH from any place, you may selectively access folders. If you already know the path to the folder you want, it is often faster to enter that folder's name then, instead of pressing RETURN, press QUESTION MARK to see the files in that folder. If, for instance, you knew that there was a disk in one of your drives named NFL, and that there was a folder on that disk named WINNERS, you could see a list of that folder's files by entering

```
/NFL/WINNERS/?
```

at the "ENTER PATHNAME" prompt.

Format of a Selector Entry

In the first column of each file's entry in the selector's list, there is either a blank space or an ASTERISK. If there is an ASTERISK, the file is locked and may not be deleted or otherwise altered until the file is unlocked. The next column of the catalog listing shows you the name of the file. After the name of the file, the file's entry in the listing shows you the number of blocks that file uses on the disk. (A block is a unit of storage that translates to 512 characters.) Next is the file's type. The filetype table, below, shows each file type available. The final column in the catalog listing is the date on which the file was last modified. At the top of the screen is the name of each column in the list and, in the far-right-hand column, the amount of free

space on the disk. The free space count is expressed as the number of blocks available on the disk. Note: in an effort to minimize extraneous information, the heading is not spoken. To examine the disk's free space, enter TEXTALKER's review mode and look at line "B". (See Appendix D" for complete details on TEXTALKER's review mode.)

Selector Commands

Once you access the selector, you have a range of commands available. You can use, in short, all the APPLE disk access commands. Following is a list of these commands. Note that when a command is specific to a program, that program's name is shown in the right column.

Selector Command Key Table

A	APPEND (ProWORDS)
A	ADJUST-FILETYPE (ProTERM)
B	BLOCK-SAVE (ProWORDS)
C	COPY
D	DELETE
E	EXECUTE
F	CAPTURE-BUFFER-OFF (ProTERM)
I	INSERT (ProWORDS)
L	LOAD (ProWORDS)
L	LOAD-MACRO (ProTERM)
M	MAKE-FOLDER
N	NAME CHANGE
O	OVERFLOW-LOAD (ProWORDS)
P	PRINT (ProWORDS)
Q	QUIT
R	RECEIVE
S	SAVE
T	TRANSMIT
U	UNLOCK (OR LOCK)
V	VIEW

Special Commands In addition to pressing ESCAPE to return to editing or terminal mode and the alphabetic disk access commands, there are command keys which perform special functions.

ARROWS The UP ARROW and DOWN ARROW keys move the selector's pointer to either the next file up or the next file down the list. If you press UP ARROW while pointing at the first file, the selector sounds a tone and moves the pointer to the bottom file in the list. Similarly, pressing DOWN ARROW while pointing at the bottom file moves the pointer to the top file.

If the list is a long one, you may choose to skip several files by holding down the APPLE key while pressing the UP or DOWN ARROW.

RETURN RETURN is the smart command. The action RETURN performs depends on the effected file's filetype. If the file to which you are pointing is a text file, pressing RETURN always loads the file into memory. If it is a folder, the folder is opened and a list of files in that folder is displayed. If the file is another system program (type SYS), that file will be executed. Be careful not to press RETURN when you are pointing to a system file unless you have already saved any text that you might want to save.

SPACE The SPACE BAR shows another list of files. The list to next display depends on which folder is open. If you are looking at the main folder of a disk, SPACE shows a list of all the disks available in your system. If a folder within the main volume is open, SPACE shows a list of the files in the folder that contains the previously open folder. The SPACE command, in short, always shows the current folder's parent folder.

SLASH The slash command is used to access the main folder of the disk in the last drive you used. There are at least two reasons for using the slash. First, it permits you to quickly access the main folder if you are currently buried several folder levels deep. The SPACE command would eventually bring you back to the disk's main folder, one level at a time, but the SLASH moves you there immediately. Second, the SLASH allows you to remove the disk in the current drive, insert another disk, and examine that disk's contents.

ESCAPE ESCAPE exits the selector and returns you to editing if you are using ProWORDS or to terminal mode in ProTERM.

Disk Access Commands Common to ProWORDS and ProTERM

Each command common to both programs is discussed in detail here. Commands specific to each program are discussed in the appropriate section of the manual.

COPY [pathname] copies the file, no matter its type, to another disk or folder. When you press either APPLE-C (or C from the selector), the program asks you to tell it the name of the file to copy. Type the name of the file and press RETURN (or just press RETURN if the selector typed the name for you).

After you type the name of the file and press RETURN, the selector asks where you want the file copied. It expects that you type the whole name all the way down to the exact file name. Note, for instance, that if you wanted to copy a file called NOTES from your disk named STUFF to another disk named IMPORTANT and you still wanted to call it NOTES, you would type the following pathnames.

/STUFF/NOTES

/IMPORTANT/NOTES

If, on the other hand, you want to call the file, when it is moved to the IMPORTANT disk, HISTORY instead of NOTES, you could type these pathnames.

/STUFF/NOTES

/IMPORTANT/HISTORY

Remember that like all commands in ProWORDS and ProTERM, you can use the COPY command without disturbing the text currently in memory.

DELETE [pathname] deletes a file or program from your disk. Once you press APPLE-D, the selector waits for you to type the name of the file to delete. If, of course, you pressed APPLE-D or D from the selector, the name automatically appears. Press RETURN to delete the file from your disk. If the file is locked, use the APPLE-U, UNLOCK, command to unlock the file before deleting it. Make sure, however, that you really want to delete the file. After all, why is it locked in the first place?

EXECUTE [pathname] executes the program whose name you supply. You should make sure, before executing another program, that you have saved any text which you have been editing. (ProTERM, if the buffer is on, automatically saves the file when the execute command is issued.) Note, only system (SYS) files may be executed. If you try to execute a file of another type, you will see the "INVALID FILETYPE" warning and will be requested to try again. When you execute a program, the program you are leaving (either ProWORDS or ProTERM) is erased from memory and the new program is loaded and executed.

MAKE-FOLDER [pathname] makes a new folder with a name that you specify. Once a folder is made, you may store other files in that folder. Say, for instance, that you have a disk named NFL. To make a folder named WINNERS on the NFL disk, press APPLE-M and type

/NFL/WINNERS

Now, to save a file named JETS in that directory, press APPLE-S and type

/NFL/WINNERS/JETS

To see all the files in the WINNERS folder, press a command - it does not matter which one because you may always exit back to where you were - then type

/NFL/WINNERS/?

If you try to make a new folder and you give it the name of a file that already exists in the open folder, you will be warned with the "DUPLICATE FILE NAME" message and requested to try it again.

QUIT ends the program.

NAME CHANGE [pathname] renames the file who's name you supply. Once you enter the name (or choose "N" from the selector) and then press RETURN, you will be asked to type the new name for the file. Type the new name and press RETURN. Next, you are asked to type the new name for the file. Type that name and press RETURN. After the renaming is complete, the program asks for the name of another file to rename. If you don't want to rename another file, press ESCAPE to return to editing or press the command you want next.

VIEW [pathname] shows you the contents of another text file on the disk without disturbing any text in memory or any text in the capture buffer. There are times when you will want to refer to another file without actually loading the file into memory. The view files command (APPLE-V) permits such an operation. Type the file name to view (or choose "V" from the selector). Once the file has been displayed, ProWORDS and ProTERM return you to the "VIEW" prompt ready to accept the name of another file to view. To return to editing or to the terminal mode, just press ESCAPE.

If, while viewing a file, you decide not to read the entire file, you may press ESCAPE to return to the "VIEW" prompt. Once at the "VIEW" prompt, press ESCAPE again to return to editing or to the terminal mode.

While viewing a file from within ProWORDS, it is possible to extract portions of the file and insert them into the current text in memory. To extract characters from the file, hold down the APPLE key as the material is displayed. The Apple's speaker clicks to help remind you that these characters are being inserted into the text.

There are two other significant keys that can be used while viewing a file. The UP ARROW key repositions the file's pointer back so that, if you missed a part of the file, you can have another chance to read or capture it. The DOWN ARROW key permits skipping through the file very quickly. This helps when the material you want to find may be deep within the file. Note that while using UP ARROW, speech is automatically restarted at a position in the file slightly before the current position. DOWN ARROW, on the other hand, is designed to let you skip through large files by leaving the speech off until you press the SPACE BAR to continue listening.

UNLOCK [pathname] either locks or unlocks the file who's name you specify. If the file is locked, APPLE-U unlocks it, if it is unlocked, APPLE-U locks the file.

COMMA [folder name] opens the named folder. Once a folder is open, you can refer to files in the folder by name only--that is you don't have to type the disk's name or the folder's name.

Filetypes

Some files cannot be loaded into the work space for editing. Remember, ProWORDS uses text files (TXT) to load and save its data. If you try to load another kind of file, ProWORDS tells you the file has an invalid file type. Some files, like those with a filetype of "SYS" are programs. Some programs with a filetype of "SYS" on your master disk are BASIC.SYSTEM, WORDS, TERMINAL, FILER, and CONVERT. To execute one of these files, press APPLE-E and type its name (or press RETURN in the selector.) Some files have a filetype of "DIR". These are directories or folders. Each folder may contain lists of other files. To see the contents of a folder, access the selector and move to its name on the catalog listing, then press RETURN. If you select a folder and decide that it is not the one you want, press the SPACE BAR; the selector shows you the folder that was active before you selected that one. (If you wish to make a new folder, press APPLE-M.)

Remember, if you don't see the file you wish to work on, press the SPACE BAR; the selector will move out one folder level (to the current folder's parent.) If the current folder is the main folder of the disk, pressing SPACE shows a list of all disks.

Filetype Table

TXT	TEXT FILE (PROWORDS DATA or ProTERM macro)
DIC	SENSIBLE SPELLER (TM) DICTIONARY FILE
BIN	BINARY PROGRAM OR DATA
DIR	SUB DIRECTORY or FOLDER
ADB	APPLEWORKS DATABASE FILE
AWP	APPLEWORKS WORD PROCESSOR FILE
ASF	APPLEWORKS SPREADSHEET FILE
LIB	ALU LIBRARY FILE
PAS	PASCAL PROGRAM
CMD	APPLESOFT ADDED COMMAND FILE
INT	INTEGER BASIC PROGRAM
IVR	INTEGER ADDED VARIABLES
BAS	APPLESOFT BASIC PROGRAM
VAR	APPLESOFT VARIABLE FILE
REL	RELOCATABLE EDASM FILE
SYS	SYSTEM PROGRAM

Common Problems

Probably the most common error new users make is to forget that the name of a disk (volume) must be preceded with a SLASH (/). Each subsequent part of the name is then separated with SLASHES. (The SLASH at the end of the pathname is optional.) You only need to

specify the entire pathname if you wish to access a file which is not in the open folder. To open another folder, press APPLE-QUESTION MARK and move to the folder you want.

When using the selector, you can save time by making sure that you have a disk in every drive. The disk does not have to be a ProDOS disk, but it should not be a blank disk.

Disk Errors

Disk errors have several causes. You may have forgotten to close the disk drive door or tried to load a file that does not exist or attempted to save a large file to a disk that is nearly full.

Here is a table of some common errors, their causes, and one's possible recourse:

ERROR CAUSE

DIRECTORY FULL There are too many files stored on the disk's main folder. (ProDOS allows only 51 files on the main folder.)

REMEDY: make other folders in which you may store more files. To make a folder, use the APPLE-M command from ProWORDS or ProTERM or use the CREATE command from BASIC. Folders are used to hold files of similar type or subject. Use this powerful feature of ProDOS to organize your data.

DISK FULL There is no room left on the disk.

REMEDY: Use another disk or delete a file that is no longer needed. It is a good idea to keep several formatted disks on hand. If you do not have a formatted disk, you may wish to temporarily save the file to the RAM volume until you can format one.

DIRECTORY NOT FOUND The specified folder does not exist.

REMEDY: Make sure you spelled the folder name correctly.

DISK NOT FOUND This error occurs when the disk you specified is not in one of your disk drives. Usually, the disk you want is there; you probably incorrectly specified its name.

REMEDY: Make sure the disk name is preceded with a slash (/). Make sure each subsequent part of the name is also separated by slashes. If typing only the file name (not preceded with the path to that file), make sure that you did not accidentally type a slash in front of the name. Remember, slash, at the beginning of a pathname, indicates that you are specifying a volume. Check the disk to make sure the disk is, indeed, in one of your drives. Be sure the drive door is shut.

DUPLICATE FILE NAME You tried to make a folder with a name that is already used for another file or you tried to change a file's name to one that is already used.

REMEDY: Choose a different name for the new folder or file.

FILE LOCKED The file is protected. Someone used the lock command expressly to protect it from accidental deletion or to keep from accidentally over-writing it.

REMEDY: Use the unlock command (APPLE-U) if the action is still desirable. Choose a different name for the file.

FILE NOT FOUND The requested file does not exist on the specified folder.

REMEDY: Check the spelling. Specify the entire pathname.

FILETYPE MISMATCH The file you specified is not the appropriate kind. If you are trying to load a file, it must be a text file (type TXT.) If you are trying to execute another program, the file must be a system file (type SYS.)

REMEDY: If you are trying to save a file, give the file a different name. If you are loading a file, it must be a text file. If you want to execute another program, it must be a system file (type SYS.) To run a BASIC program, execute BASIC.SYSTEM then run the program.

SYNTAX ERROR Unrecognized command.

REMEDY: Make sure the command is typed correctly. Remember that no spaces or punctuation may be included in a file name.

Appendix D: Speech

This disk contains two speech programs, TEXTALKER for the Echo synthesizer from Street Electronics Corporation and ProSCAT for the SlotBuster synthesizer from RC Systems, Inc. ProWORDS, ProBRAILLE, and ProTERM provide you complete access to all the commands and features unique to each of these programs. The tables that follow provide brief command summaries for each. They are not intended to substitute for the manual that came with the program, but the tables provide a handy place to look up specific commands.

The following commands all start with CTRL-E. When you see "n", substitute a number in the range suggested for each command. Note that when you type one of the following commands, the speech program absorbs the command so the applications program you are using never even knows you altered your speech parameters.

TEXTALKER Commands

A puts TEXTALKER into "all" punctuation mode.
B sets both screen and speech output.
C sets compressed speech mode.
nD sets the amount of delay between words. The range of n is from 0 to 15.
E sets expanded speech rate.
nF sets the pitch level that is flat. Use numbers from 1 to 63.
I sets up columns for line review mode. See the TEXTALKER User's Manual for details.
K sets TEXTALKER to spell words that are capitalized.
L sets letter mode.
M sets "most" punctuation.
N turns off the CTRL-E K command.
O sets output only to the screen. (It turns off the speech.)
nP sets the pitch. Use numbers from 1 to 63.
nR sets the repeat filter. Use numbers from 0 to 99.
S sets some punctuation mode.
T sets talk only mode.
nV sets the volume. Use numbers from 1 to 15.
W sets word mode.

ProSCAT Commands

nA sets articulation. Use numbers from 1 to 8.
nB sets buffering mode. Use 0 or 1.
nC sets the level of character punctuation. Use numbers from 1 (least) to 3 (most).
nD sets digits mode. Use 0 or 1.
nE sets output mode. Use numbers from 1 to 4.
F sets frequency. Use numbers from 230 to 235. See the SlotBuster User's Manual.
H sets hard copy slot. Use numbers from 1 to 7.

I sets emulation mode.
P sets pitch.
S sets speed. Use numbers from 1 to 16. (Only 12-15 usable.)
T sets text punctuation level. Use numbers from 1 to 3.
V sets volume. Use numbers from 1 to 15.
W sets word delay. Use numbers from 1 to 15.

Appendix E: MODEM COMMANDS

Hayes and compatible modems use the commands that follow. Your modem may require that you issue these commands in upper case letters. Check your modem's instruction manual to make sure.

Your modem is always in one of two states, command or online. Note that you issue the attention commands to your modem while it is in the "command" state. Once you issue the dial command and connect to another system, the modem enters online state and simply transmits data between your computer and the remote system. You may regain control of command mode by issuing three "+"s while you are connected. See more about the escape code command in the table below. Note, too, that some "Hayes compatible" modems do not use the escape code (+++) to access command state but, instead, disconnect the remote system by hanging up the phone. Check your modem's manual to see how your modem handles the escape code.

The smart modem waits until you press RETURN before acting on the command. When you press RETURN, after typing a command, the smart modem tries to execute the command then returns a result. In most cases, the result is "OK". See the table below for each command's result code:

Table of Modem Commands

COMMAND ACTION

AT ATTENTION. The "AT" command means "attention" and precedes most of the following commands. Exceptions are noted. The "AT" command must follow a carriage return. Do not press RETURN after typing "AT". Instead, type the specific command code.

A/ REPEAT THE LAST COMMAND. The A/ command executes the last command issued. This is most often used to redial a number after using the dial (D) command and then getting a busy signal. In such a case, you may press any key to hang up the phone then issue the A/ command to redial the number.

D DIAL. The "D" command dials the phone. The syntax is: ATD [number to dial]. You may use punctuation in the number. It is, except for the ",", ignored. The "," tells the smart modem to wait two seconds before dialing the next number. This permits you to use the smart modem on switchboards that require you to dial a "9" then wait for an external dial tone.

COMMANDS USED IN CONJUNCTION WITH D: You may use "P" in the dial command to indicate pulse dialing. Likewise, enter "T" for tone dialing.

RESULT CODES: The result code depends on the response of the remote system after the dial command is executed. If the remote system answers and sends a carrier signal, the smart modem gives the result code "CONNECT" and enters the online state. If the remote system does not send a carrier within a certain time, 30 seconds by default, the smart modem hangs up the phone and returns the "NO CARRIER" result code.

T TONE DIALING. Using the "T" command on a line by itself (ATT), causes the smart modem to dial all subsequent numbers with tones.

P PULSE DIALING. The "P" command on a line by itself (ATP) causes the smart modem to dial all subsequent numbers with pulses.

M0 MONITOR SPEAKER OFF. The "M0" command completely disables the modem's monitor speaker.

M1 SPEAKER ON FOR DIALING. The "M1" command causes the smart modem to leave the monitor speaker on during dialing and until a carrier is detected. Once the carrier is detected, the speaker is disconnected.

M2 SPEAKER ON ALWAYS. The "M2" command causes the smart modem to leave the speaker on at all times. If you are transferring data at 300 BAUD, this command allows you to hear the data transfer. Note that at 1200 BAUD, nothing but a harsh hiss can be heard. You may need to adjust the volume of the speaker with the volume knob on the back panel of the modem.

Command Registers

The smart modem utilizes 10 command registers which control several aspects of your command sequences. These are called the "SET" registers and may be set by you. The set command is issued by entering the attention command (AT), the set command (S), the number of the register to set, an "=", and the value to which you wish to set that register.

A typical set command sequence might look like this:

```
ATS0=1
```

The 0 command register determines the number of the ring on which your modem answers the phone. In this example, we set the value to "1" so the modem will answer the phone on the first ring.

You may also examine the value of any register. If, for instance, you want to see the value of the S0 register, enter this command:

```
ATS0?
```

If you had just turned on the modem, the smart modem would return "0".

Table of Command Registers

REG DEFAULT FUNCTION

S0 0 NUMBER OF RINGS BEFORE SMART MODEM ANSWERS THE PHONE. Note that, if the S0 register is set to 0, the smart modem does not answer the phone. The range for the S0 register is from 0 to 255.

S1 0 RECORDS THE NUMBER OF TIMES THE PHONE HAS RUNG.

S2 43 ASCII CODE OF THE ESCAPE CHARACTER. ASCII 43 is the "+" character.

S3 13 ASCII CODE OF THE END OF LINE CHARACTER. ASCII 13 is the code for the carriage return character.

S4 10 ASCII CODE OF THE LINEFEED CHARACTER.

S5 8 ASCII CODE OF THE BACKSPACE CHARACTER.

S6 2 TIME (IN SECONDS) TO WAIT, AFTER PICKING UP THE PHONE, BEFORE ACTUALLY DIALING THE FIRST NUMBER OF A PHONE NUMBER.

S7 30 NUMBER OF SECONDS, AFTER DIALING THE NUMBER, THE SMART MODEM WAITS FOR A CARRIER SIGNAL.

S8 2 LENGTH OF TIME (IN SECONDS) TO PAUSE FOR THE COMMA COMMAND.

S9 600 LENGTH OF TIME (IN MILLISECONDS) REQUIRED BEFORE THE SMART MODEM RECOGNIZES A RETURN CARRIER SIGNAL.

S10 700 TIME (IN MILLISECONDS) THAT LAPSES AFTER LOSS OF CARRIER, BEFORE THE SMART MODEM DISCONNECTS AND RETURNS TO LOCAL STATE.

S11 70 TIME (IN MILLISECONDS) OF THE DURATION AND SPACING OF THE TONES FOR TOUCH-TONE DIALING.

S12 1 TIME (IN SECONDS) OF THE ESCAPE GUARD TIME. This means that if this register is set to "1", its default, you must wait one second before and after issuing the escape code (+++). This permits transmission of the escape sequence yet still allowing a way to gain control of the modem's local state.

Appendix F: TROUBLESHOOTING

The series of steps required to transmit and receive data between your system and a remote computer is both complex and full of potential trouble spots. This appendix outlines common problems, their symptoms, and possible solutions.

WHEN I TYPE "ATT", MY MODEM DOES NOT RESPOND WITH "OK".

The modem is not recognizing your keystrokes. Make sure the modem is a smart modem and is able to recognize this command. Check to see that the modem is plugged in and is turned on. Make sure the cable from the Super Serial Card to the modem is connected properly and securely. Try typing the command in upper case letters. Make sure the jumper block on the SSC is set so that the triangle points toward the word MODEM. Make sure the SSC's switches are set as described in the "Introduction" section of this manual. Examine the dip switches on the modem; set them as they came from the factory.

WHEN I TYPE MODEM COMMANDS, THEY DO NOT PRINT ON THE SCREEN.

Look at the dip switches on the modem. They should be set as they were when the modem came from the factory. Check the modem's user's manual to obtain the factory settings. Some modems require that you enter the echo command (ATE1) before echoing keystrokes is enabled.

I GET NOTHING BUT GARBAGE WHEN I CONNECT TO A REMOTE COMPUTER.

The parameters are probably set incorrectly. Check with the system's operator to get the correct data format, stop bits, and baud rate; then adjust, with the APPLE-P command, the parameters so that ProTERM's parameters match those of the remote system.

WHEN I COMMUNICATE WITH MY FRIEND'S COMPUTER, I DO NOT SEE WHAT I TYPE.

Select "Echo Keystrokes" from ProTERM's parameters menu.

When I communicate with a remote computer, I loose characters, especially at the beginning of a line.

Look at the DIP switches on your Super Serial Card. The sixth switch on the second bank should be on. If this switch is already on, contact your dealer.

I loose characters in the middle of long files.

Most likely, the remote system does not respond to the X-on X-off protocol. If it is possible, adjust the remote system's parameters to support this handshaking protocol. If the remote system supports the XModem file transfer protocol, use XModem to transfer the file.

Some users have reported problems when using the TimeMaster (TM) clock card. Any problems associated with this card can be corrected by setting the switches on the card so that the time is not able to be set. We are unsure of the cause of this problem but know the fix.

Appendix G: DAVEX, Talking Command Interpreter for ProDOS

Your master disk from APH contains a limited version of DAVEX. DAVEX is a powerful command shell for ProDOS. It lets you perform several useful disk commands like copying files, deleting files, or starting other programs. When you do start another program, you automatically return to DAVEX when you exit that program. See the end of this document for details on getting a full fledged copy of DAVEX.

This document describes DAVEX as it is implemented on the MicroTalk/APH master disk. It is not the complete DAVEX documentation. You can find that in the file DAVEX.DOC on your disk.

When you boot your disk, DAVEX is automatically started. DAVEX shows you a short list of programs. You can start using one of those programs by typing its name or by typing the first letter of its name then pressing RETURN. That list looks like this:

```
Help Words Spell Term Filer Grade2 BASIC
```

Note that not every program on the list is installed. You can add the other programs later if you wish, or you can add programs that aren't listed.

In addition to the line of program names, the initial screen shows you the name of the open folder (or directory.) It is /USER/UTIL. For an understanding of folders, think of your disk as a drawer in your file cabinet. You can consider the drawer as a folder itself--the main folder on that disk--it holds other folders and files. Each folder can contain several files or more folders. The name DAVEX shows you has both the drawer's name (/USER/) and the folder's name (UTIL/). Notice that the drawer's name begins with a SLASH. That lets you know the name refers to the disk's main folder. Notice too that the name of both the drawer and the folder ends with a SLASH character. The SLASH is used to separate the parts of the name. The final SLASH is optional. You can open another folder simply by typing its name.

The disk shipped from APH doesn't have any more folders on it, but you can create new folders if you wish. You do this by typing the word "create" and the name of the new folder. If, for instance, you want to create a new folder inside your UTIL folder named NOTES, type "CREATE NOTES". Before you do that, though, think about where you want that new folder located. You would probably rather have it in the main drawer rather than inside another folder. Remembering that the name of your drawer is /USER, you can create the NOTES folder in one of two ways. Open that drawer by typing /USER, then create the folder by typing "CREATE NOTES". You can also create the new folder without opening the folder in which

it is to be created. You do that by typing both the name of the drawer and the name of the new folder like this:

```
CREATE /USER/NOTES
```

You can have a folder named NOTES inside your UTIL folder and have another one named NOTES in your /USER drawer. To open the folder, just type its name. In other words, to open the NOTES folder in the UTIL folder, type "/USER/UTIL/NOTES". To open the folder in the /USER drawer, type "/USER/NOTES". If you want to open the NOTES folder when you already have the /USER folder open, just type NOTES. Similarly, if you already have the /USER/UTIL folder open and you want to open the /USER/UTIL/NOTES folder, just type NOTES. If you forget what folder is open, just press the RETURN key. That makes DAVEX show you two things. First, DAVEX displays the current time and date. Then, it shows you the name of the open folder.

Once you open a folder, you might want to list its contents. You can do that by typing the letter "L" and pressing RETURN. If you haven't put any files in either of your two new folders, they will obviously be empty. To see some folders that aren't empty, try opening the /USER drawer and the /USER/UTIL folder and listing their contents. The /USER drawer's contents looks something like this:

```
/user/ 7 files
```

```
name type
prodos sys
start.system sys
util dir
updates txt
startup bas
term.updates txt
davex.info txt
```

Notice that, in addition to the folder's name and the number of files shown at the beginning of the listing, there are two columns of information. One is the name of the file and the other is the file's type. There are four different types of files in the /USER folder. They are SYS, TXT, DIR, and BAS. Each of these types are important, but TXT and DIR are of most concern, now. SYS and BAS files are programs. Files labeled with the type DIR are folders. Notice that UTIL's type is DIR. DIR is short for directory. You can think of it as a folder but the computer likes to call it a directory. The currently opened folder is also often called the current prefix. The word prefix is used because it lets you refer to files in the open folder without typing the folder's name. You can also refer to other files on the disk in other folders by using the folder's name and the file's name. This combination of names is often called the complete path or pathname to a file. It is called a path because it completely describes where the file is

located--that is it specifies the disk, any folders on that disk, and the specific file on that disk.

TXT files are documents that you write with the word processor. They contain text. (That's what the "TXT" stands for.) The TXT files in the /USER folder are files that I wrote. You can read them with WORDS and your speech synthesizer. The one named UPDATES contains information about your master disk that has changed since the manual's publication. The one called DAVEX.INFO contains the text of the material you are reading now. If you want to use WORDS to read it, open the /USER folder by typing "/USER" and pressing RETURN, then start the word processor by typing "WORDS DAVEX.INFO". Notice that is the command "WORDS", a SPACE, and the name of the file, "DAVEX.INFO". Your disk spins as DAVEX starts the word processor, then it spins a little more as the word processor opens the DAVEX.INFO file. When all this spinning quits (in about ten seconds), press your computer's DOWN ARROW key to read the text. When you want to quit reading the text, press APPLE-Q. That's the word processor's command to quit. When you press APPLE-Q, the word processor wants to make sure that you didn't accidentally press APPLE-Q, so it asks, "ARE YOU SURE?" Press the "Y" key to tell WORDS, "YES". When you do, you'll see the familiar menu line and the name of the open folder.

Note that these instructions assume that you're using the 3.5 inch disk version of the /USER disk. If you use 5.25 inch disk drives, The menu line that is shown when you start your system also tells you to flip over the disk. (There isn't room on one side of a 5.25 inch disk to hold all the files.) After you've flipped over the disk, insert the second disk in your second disk drive. That's where the DAVEX.INFO is located on the 5.25 inch disk version of the Master disk set. The name of that disk is /HELPPDISK. To read the file with ProWORDS,, open that disk's main folder by typing /HELPPDISK. Remember that initial SLASH. Then, type "WORDS DAVEX.INFO". To read the DAVEX.INFO file with the word processor on a single 5.25 inch disk, type "WORDS" then press RETURN. When the program starts, it'll show you a title screen that contains copyright information, APH's address, and the version number of the program. The title screen also tells you to press the SPACE to continue. Do it. Then insert the second disk and press APPLE-L. (APPLE-L is a command that tells the word processor to "load" a file into its work space.) When you press APPLE-L, ProWORDS asks for the name of the file to load. Type DAVEX.INFO and read it by using the DOWN ARROW key.

If you open the /USER/UTIL folder and list its contents, you'll see something like this:

```
/user/util/ 23 files
```

```
name type
davex sys
```

```
aliases txt
basic.system sys
startup bas
proscat bin
textalker bin
pt.obj bin
sysalias bin
init bin
comp bin
config bin
menu.txt txt
autoexec txt
help dir
floppy txt
filer sys
words sys
words.macros txt
term sys
term.macros txt
brl sys
tbl2 txt
b.tbl2 txt
```

You can see that there are a lot of files in the UTIL folder. In addition to the SYS, DIR, BAS, and TXT files, the UTIL folder has some files with a type of BIN. Those are files that hold binary programs. It is difficult to tell what a binary file does just by looking at its name. The files, ProSCAT, TEXTALKER, and PT.OBJ all contain programs that make your Apple talk with either the SlotBuster or the Echo synthesizer. The rest of the BIN files are external DAVEX commands. More about them shortly.

The UTIL folder is a special place. DAVEX always looks there to find programs that you want to use. (You can also make DAVEX look in other places by opening the folder where the program is located, then typing its name.) The nice thing about the UTIL folder, though, is that DAVEX always looks there no matter what folder is opened. If you want to add more programs to the UTIL folder, you can do so by copying the program there. If, for example, you buy the SENSIBLE SPELLER, you'll want to copy that program and possibly its dictionary to the UTIL folder. If you have SENSIBLE SPELLER, you'll know that the name of its disk (or drawer) is /SENS. The name of the spell checker program itself is SPELL. You can copy the program to your UTIL folder by following these steps. Even if you don't have the speller, you might want to follow the discussion; the process of copying files is the same for any file. Note, however, that 5.25 inch disk users will not be able to copy the speller to their disk. Again, following the discussion should help clarify the command procedure for this and other DAVEX commands.

The COPY command has three parts. First, the command itself is typed. Next, you tell DAVEX where to find the files to copy. Finally, you tell DAVEX where to put those files. Each part of the command is separated from the rest of the parts with a single SPACE. There are several ways to copy the speller files to the UTIL folder. Here is one.

1. Open the /SENS directory by typing /SENS.

2. List the contents of the /SENS folder with the "L" command. You'll see four type SYS files. You don't need the two named PRODOS and BASIC.SYSTEM. You want the two named SPELL and SETUP.SPELL. SPELL is the spell checker program, and SETUP.SPELL is the program used to configure SPELL.

3. Copy the two programs to the UTIL folder. Do this with DAVEX's COPY command. You use the copy command by typing the word "COPY", the name of the file, and the name of the folder where you want to copy it. Copying SPELL, for example, is done with the command "COPY SPELL /USER/UTIL". Copy SETUP.SPELL with the command "COPY SETUP.SPELL /USER/UTIL". Note that these examples both assume that you've opened the /SENS folder. If another folder were open, you'd have to use a command like "COPY /SENS/SPELL /USER/UTIL". But, if the /USER/UTIL folder were the folder you had open, you could do it with the command "COPY /SENS/SPELL". In other words, DAVEX adds the name of the open folder to your specifications if you give only the file's name.

4. Once you copy both programs to your UTIL folder, start the SETUP.SPELL program with the command "SETUP.SPELL" to configure the speller. See your speller's instruction manual for complete details on this process.

What happens if you want to open a folder or drawer and you don't know the name of it? There are several ways to open folders with names that you don't know. If you know the slot and drive number of the disk drive containing the disk of interest, you can tell DAVEX to open that drawer by using that information. If, for example, you know that you have SENSIBLE SPELLER's dictionary disk in drive one of your 5.25 inch drives, and your 5.25 inch disk drives were connected to slot 6 (the normal place for them), you could tell DAVEX to open that folder by typing ".61". That is type a PERIOD, the slot number, and the drive number with no SPACE characters between them. If you typed ".61" with your dictionary disk in that drive, DAVEX would show you the name of the newly opened folder with a response like "/D1". (/D1 is the name of the disk that holds the dictionary.) To see the files in that folder, use the "L" command again.

You can use the slot and drive method of identifying folders in the copy command. In the previous example, you saw that no matter what folder you have open, you can copy the speller to your util

directory by specifying the drawer's name. You used the command "COPY /SENS/SPELL /USER/UTIL". If the disk containing the speller were in slot 6 drive 1, the command "COPY .61/SPELL /USER/UTIL" also works. Further, if your /USER disk were in slot 5, drive 1, the normal place for a 3.5 inch disk drive, you could use the command "COPY .61/SPELL .51/UTIL". If the open folder were already /USER/UTIL, you could copy the speller with the command "COPY .61/SPELL".

This example shows you how to individually copy files. DAVEX also supports a handy feature that lets you copy groups of files at a time. You can copy all the files that match a pattern by using one of two "wild card characters." Wild cards are characters that stand for a group of characters. There are two wild cards for a reason. One of them, the QUESTION MARK makes DAVEX ask you before it copies a file that matches your specifications. The other, the EQUALS sign, copies the specified files without asking. In the example with the /SENS disk, you can tell DAVEX to look at each file and ask you to confirm by pressing the "Y" key. (The "Y" stands for "YES.") Type "COPY /SENS/? /USER/UTIL/?" to make DAVEX show you the name of each file on the disk and wait for you to answer with "Y" for "Yes, copy the file" or "N" for "No, don't copy the file." Although you wouldn't really want to do it with this disk, replacing the ? with a = makes DAVEX copy every file.

Using the wild card character as part of the name lets you be more selective about which files to copy. You can copy files that begin with "S" by using the command "COPY /SENS/S= /USER/UTIL/S=".

DAVEX uses two kinds of commands. The ones discussed so far are internal commands. That means they're built in. The other type of commands are external. External commands are ones that DAVEX loads from the disk.

Here is a list and brief explanation of some of DAVEX's internal commands.

\$ shows you information about how to purchase the full fledged version of DAVEX.

Bye makes DAVEX disconnect itself from the system. There is no reason to do this though, so it is recommended that you not use DAVEX's bye command. If you want to use another program, just type its name.

Version displays the program's version number.

Alias lets you make one command work with another name. The "L" command is an alias for the cat command with the "short" option.

Config lets you specify information about your system.

Como lets you capture the output from a command into a file.

Exec makes DAVEX take its commands from a file rather than the keyboard.

Boot lets you boot your machine.

Up closes the opened folder and opens the folder containing the currently opened folder.

Top opens the drawer or the main folder of a disk.

Help provides online information about any command.

Online shows you the names of all the disks in your drives.

Cls clears the screen.

Type shows you the contents of a text file.

Rename changes the name of a file.

Create creates a new folder.

Delete deletes a file or folder.

Lock locks a file.

Unlock unlocks a file.

Scan lets you add more folders to DAVEX's search for commands.

Cat shows you a catalog of a folder.

Info gives you information about a file.

Update lets you copy only files that have changed since the last use of this command.

Copy copies files.

Move moves files.

Dev lets you add or remove devices from your ProDOS scan list.

Eject ejects a 3.5 inch disk from its drive.

Here are some of the external commands that DAVEX uses. Note that not all these commands are included in the limited version of DAVEX on your /USER disk.

Init initializes a disk.

Comp shows settings and lets you adjust the settings of the //gs control panel.

Vstore stores the entire contents of a volume into a single file. This is useful for RAM disk users.

Vrestore restores a file packed with the Vstore command.

Deschw describes your computer system.

Setdate lets users without a clock set the date and time.

GSbuff sets the size of the //gs's built-in printer buffer.

When you give DAVEX a command, you often include extra information about how you want the command to work. In the case of the copy examples previously discussed, the extra information was the name of the file to copy and the place to copy it. You can add even more information to the copy command to make it behave differently. You do this by adding options to the command. Options are specified by typing a DASH (-) followed by a letter that denotes the option. The copy command has several options that can be used. You can, for example, tell DAVEX to delete the file from the original disk after it is copied. This is done with the D option. A copy command that copies the file TEST from the /JUNK disk to the /GOOD disk, then deletes the file from the /JUNK disk looks like this:

```
COPY /JUNK/TESTFILE /GOOD -D
```

Another example of a useful option can be seen with the CAT command. The command "CAT -T" shows you the catalog of a folder like the normal CAT command does, but instead of showing you only the name of folders in the current folder, it shows you all the files in all the folders. The S option is also a useful option for CAT. As mentioned, the "L" command is an alias for "CAT -S" which is a shorthand form of the normal CAT command. If you use the normal cat command, you get more information than you do with the S option. In addition to the name of the file and its type, you get the size of the file, the date and time of its last modification and its protection status. (A files protection status tells you if a file is locked.)

You can combine options. The command, for instance, "CAT -ST" shows you a short form of a tree catalog. Using the CAT -ST command on the /USER disk results in a display like this:

```
/user/ 7 files
```

```
name type
prodos sys
start.system sys
```

```

util dir
>
  davex sys
  aliases txt
  basic.system sys
  startup bas
  proscat bin
  textalker bin
  pt.obj bin
  init bin
  conp bin
  config bin
  menu.txt txt
  autoexec txt
  help dir
>
  indexed.help $00
words txt
term txt
spell txt
filer txt
basic txt
grade2 txt
<
  floppy txt
  filer sys
  setdate bin
  words sys
  words.macros txt
  term sys
  term.macros txt
  expand.help txt
  tbl2 txt
  b.tbl2 txt
  brl sys
<
updates txt
startup bas
term.updates txt
davex.info txt

```

Now that you know the general way DAVEX works, use DAVEX to tell you more about itself. Type a QUESTION MARK and DAVEX shows you a list of its internal commands. To get more specific information on a command, type the QUESTION MARK followed by the name of the command to read about. If, for example, you want more information on the copy command, type "? COPY".

The help is formatted in a way that helps you determine what options are allowed for each command and which options are required. Each help display starts with the command's name and a brief description of what it does. Next, the display tells you the

exact syntax of the command including required and optional parts. Notice in the sample help display from the copy command that follows that parts of the command that are required are enclosed in < and >. Optional parts of the command are delineated with [and]. After the syntax of the command is shown, each help display shows you examples of how the command is used. This section of the display is shown with "EX:" which stands for "examples". The examples show different ways to use the command with different options.

After the examples, the help display gives more detail about exactly what the command does and how options effect it. Also, any general notes of interest about the command are given here.

After the detailed explanation, the help display shows each option and explains its purpose.

After the explanation of the options, more specific notes of interest and special features and characteristics of the command are given.

Finally, the help display shows you more places to look for additional or related information.

Here is what the help display from the copy command looks like. (It was imported into this document by using the command "COMO <pathname>;? COPY").

copy -- copy files or directories

syntax: copy <pathname1> <pathname2> [-f] [-d] [-b]

ex: copy thing /otherdisk/blah
copy this /backup
copy %= /backup/= -fb
copy /system.disk/system /ram5/system

Copies files from first path to second path. Unless -f (force) is given, ask for permission to replace the old copy. If the old file is LOCKED (access is not "rwnd"), copy asks for permission even if you use -f.

-d deletes the original after a successful copy.

-b clears the backup bit of the original after a successful copy.

If the first pathname refers to a file and the second pathname refers to a directory, 'copy' puts the file INTO the directory. (It does not try to replace the directory with the file.)

If you omit the second pathname completely, 'copy' behaves as though you supplied the current prefix. Result: "copy pathname"

copies a file into the current directory, using the original file's filename.

If the first pathname refers to a DIRECTORY, the directory and everything in it (files and/or directories) is copied to the second pathname. The second pathname must not refer to an existing directory ("ProDOS error: duplicate file" will result).

'copy' will copy sparse files, but the new files will no longer be sparse.

(see also: update)

Note that 5.25 inch disk users need to have the second disk in a drive when the ? command is used with a command name. All the information for each command is stored on that disk in a specially compacted file called INDEXED.HELP. If you prefer reading about all the commands at once in your word processor, you can use an exec file located in your UTIL folder to unpack all the help files. Note, however, that the size of the information will expand, so you'll need some extra disk space. To use the exec file to expand the help into a normal text file, follow these steps.

1. Place a blank disk in drive 1. For the sake of this example, it will be assumed that the disk is in slot 6, drive 1, but it could just as well be in slot 5.

2. Initialize the disk with the command "INIT .61 EXPANDED.HELP. This initializes the disk and gives it the name EXPANDED.HELP.

3. Open the newly initialized disk's main folder by typing either "/EXPANDED.HELP" or ".61".

4. Type the command "EXEC /USER/UTIL/EXPAND.HELP". This is a command that makes DAVEX take its commands from the file EXPAND.HELP in the /USER/UTIL folder. That file uses the COMO command along with the help command for each command topic to make a large file named HELP.FILES" in the open folder or, in this case, on the /EXPANDED.HELP disk.

5. Use your word processor to load and read the expanded help files.^[You can do this with the command "WORDS HELP.FILES" if the /EXPANDED.HELP folder is already opened or with the command "WORDS /EXPANDED.HELP/HELP.FILES" if it isn't.

The implementation of DAVEX on the /USER disk was constructed with the following techniques. First, the file START.SYSTEM was created using DAVEX's SYSALIAS command. START.SYSTEM does nothing more than start BASIC.SYSTEM from the UTIL directory. When BASIC.SYSTEM starts, it executes the BASIC program called INSTALLER which is located in the /USER folder. INSTALLER first sets the open folder to /USER/UTIL, then it looks to see what kind of speech synthesizer you are using. If you have a SlotBuster connected, the INSTALLER

program loads the ProSCAT program and loads the aux rules from the file RULES.COM. If you have an Echo speech synthesizer, INSTALLER gets the speech going for that device. If you have both a SlotBuster and an Echo, the program tries to use the SlotBuster. It will use the Echo under two conditions; (1) if you had previously been using TEXTALKER or (2) if you press the "T" key while the system is starting. If you had been using TEXTALKER and you want to switch to ProSCAT, press the "P" key while the system starts.

Finally the INSTALLER program runs DAVEX.

When DAVEX starts, it always looks for a file called AUTOEXEC. If it finds the AUTOEXEC file, it automatically EXECs it. (The AUTOEXEC file is a text file of commands.) The AUTOEXEC file on your UTIL folder looks like this:

```
type %menu.txt
```

The "type" command is used to display the contents of a text file. It is used in the AUTOEXEC file to show you the menu. You can change the appearance of the menu by editing the file MENU.TXT.

DAVEX is copyrighted (C) 1988 by David A. Lyons. The limited version included on the MicroTalk/APH disk is licensed for use only with ProWORDS, ProBRAILLE, and ProTERM. The complete Davex package costs \$25. Type a "\$" at the Davex prompt to get complete information about how to register a full version of Davex.

27-10-1968

